Heft 2 März/April '87 DM 6.– ÖS 50 SF 6.–



Das unabhängige Magazin für alle Ataris

MÄRKLIN DIGITAL
- Modelleisenbahn am Atari ST

ST-TRICKS
- Directory in GFA-Basic
- Dateiverschlüsselung
- Memorix

AUFRÜSTUNG FÜR 8-BIT-ATARIS
- 320 KByte im Eigenbau

Games
- Spielreviews in Hülle und Fülle



Editorial

Lieber Leser,

die Spannung in der Redaktion war zwar groß, wie die Nr. 1 des **ATARI-*magazins*** bei Ihnen ankommen würde, aber wir wurden mit so vielen Vorschußlorbeeren bedacht, daß wir eigentlich sicher waren, mit unserem Konzept richtig zu liegen. Es hat sich nun gezeigt, daß wir mit dem **ATARI *magazin*** – das ja alle Atari-Computer berücksichtigt – voll in's Schwarze getroffen haben. Denn auf diese Zeitschrift haben einfach viele Atari-Anwender gewartet.

Mit dem **ATARI*magazin*** wollen wir Ihre Zeitschrift machen. Und da wir ausdrücklich ein Magazin für alle Ataris herausgeben, werden wir uns bemühen, immer alle Atari-Computer angemessen zu berücksichtigen. Das gilt sowohl für 8-Bit- als auch für 16-Bit-Maschinen, und wenn die Bits mehr werden, wird das **ATARI*magazin*** ebenfalls mit von der Partie sein. Es liegt also auch an Ihnen, das **ATARI*magazin*** zu *Ihrer* Zeitschrift zu machen. Nutzen Sie es als Forum, um mit anderen Atarianern in Kontakt zu kommen. Wir öffnen Ihnen unsere Seiten für Probleme, die Sie nicht lösen können, die aber andere Leser bereits abgehakt haben. Stellen Sie Ihre alltäglichen Problemlöser unseren Lesern zur Verfügung, denn oft sind es die kleinen Dinge, die einen schnell weiterbringen.



Peter Finzel

Natürlich stehen Ihnen auch unsere Mitarbeiter für Fragen und Anregungen zur Verfügung. Dazu zählen Spezialisten der ersten Garnitur wie *Peter Finzel,* dem keine Nuß zu hart ist. Die Atari-Computer bergen für ihn keine Geheimnisse mehr. Er ist ein professioneller Programmierer, aus dessen Küche manch hervorragendes Spiel und viele unentbehrliche Anwendungen stammen. Den Lesern der Zeitschrift *Computer Kontakt* ist "Peter's Assemblerecke" schon lange als Leckerbissen bekannt. Und wenn er Sie im **ATARI *magazin*** in die Programmierung von Spielen einführen will, weiß er garantiert, wovon er schreibt.

Wir wünschen Ihnen beim Lesen viel Spaß und keinen Error am Computer.

Ihre Redaktion vom
ATARI*magazin*

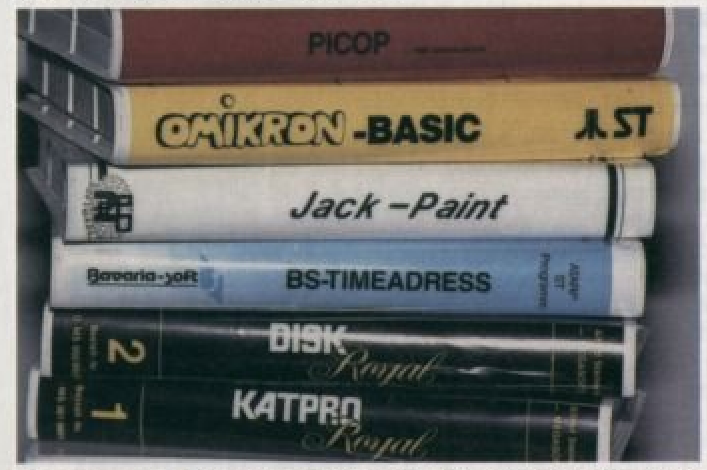
Werner Rätz



Mah Jongg – die 3000 Jahre alte Leidenschaft der chinesischen Seeleute, Soldaten und Könige wird mit "Shanghai" neu belebt. Das Spiel bringt Ihnen die Faszination des alten China auf den ST. Aber lesen Sie selbst.



Beliebige Bilder auf den eigenen Monitor zu zaubern, ist eine feine Sache. Das Digitalisieren ist glücklicherweise keine Geheimwissenschaft mehr, seitdem die entsprechenden Geräte preiswert auch für Homecomputer angeboten werden. Mit dem "Videomeister" bringen Sie den Saurier in Ihren Computer.



Die mageren Software-Zeiten für den ST sind nun endgültig vorbei. Ein nahezu unübersehbares Angebot an guter Software kommt laufend neu auf den Markt. Das ATARImagazin gibt eine Orientierungshilfe. Wir stellen Ihnen die Neuheiten vor.

INHALTSVERZEICHNIS

Das ATARI *magazin* schafft Kontakte!

Wer sich mit seinem Computer beschäftigt, möchte auch Kontakt zu anderen Anwendern. Hier bieten sich die zahlreichen User-Clubs und Benutzergruppen an. Diese Clubanschriften, Neugründungen, Termine, Nachrichten und ander Informationen aus der Szene wollen wir an unsere Leser über diese Kontaktseite weitergeben. Ausführliche Clubvorstellungen sind ebenso möglich, wie Kurzinfos, Änderungen oder Kontaktgesuche.

Wer also einen User-Club leitet oder gründen will, wer Kontakte zu anderen Computerfreaks sucht oder wer besondere Aktivitäten melden kann, sollte uns schreiben.

Unsere Anschrift:
ATARI *magazin*
Postfach 1640, 7518 Bretten

Mainz

Wir sind ein noch junger Computer-Club, der sich ausschließlich mit den Atari-ST-Computern beschäftigt. Unser Club sucht noch Mitglieder, die ebenfalls einen ST besitzen. Wir beschäftigen uns z.Zt. sehr stark mit GFA-Basic und C, wobei wir in C einen Kurs abhalten wollen. Aber auch Anhänger anderer Sprachen sind willkommen. Auch die Hardware kommt bei uns nicht zu kurz, wie z.B. der Selbstbau von Floppy-Stationen usw. Eine Mailbox ist eines der nächsten Ziele, eine Clubzeitschrift soll es allerdings nicht geben.

Jürgen Wirth
Weintorstraße 17
6500 Mainz
Tel. 06131/220372

Solingen

Im Umkreis von Solingen, Langenfeld und Leverkusen-Opladen wollen wir einen Atari-Club gründen. Geplant ist eine vierteljährlich erscheinende Clubzeitschrift sowie Clubtreffen usw. Software ist ausreichend vorhanden, eine Altersgrenze gibt es nicht. Wir wollen Kontakte vermitteln und Erfahrungen austauschen. Der Clubbeitrag beläuft sich auf 12.-DM im Jahr.

Michael Eckert
Am Weiher 3
5653 Leichlingen 2
Tel. 02174/30523

Stadtallendorf

Wir sind ein Atari-Club wie jeder andere auch, haben zur Zeit 10 Mitglieder und arbeiten mit der 8-Bit-Serie von Atari, einem 1050-Laufwerk und dem 1029-Drucker. Es gibt alle 3 Monate eine eigene Clubzeitschrift mit Listings, Basteleien und kostenlosen Kontaktanzeigen. Eine eigene Software-Bibliothek ist vorhanden. Anfragen aus dem ganzen Bundesgebiet sind willkommen. Ein Clubbeitrag wird nicht erhoben.

Thorsten Pontow
Ziegelweg 13
3570 Stadtallendorf

Pforzheim

Der 1. ATARI Club Pforzheim sucht noch Mitglieder. Clubbeitrag gibt es keinen, eine Clubzeitschrift ist in Arbeit. Wir haben zur Zeit ca. 7 Disketten voll mit Utilities und Public-Domain-Programmen. Sie sind für Mitglieder kostenlos. Wir freuen uns auch, wenn andere Clubs sich bei uns melden.

1. A.C.P.
Karolj Nad
Maximilianstraße 153
7530 Pforzheim
Tel. 07231/465156

Aschaffenburg

Gegründet im Oktober 1983 von einer kleinen Schar C-64-Besitzer, wandelte sich der Club im Laufe der Zeit wie die Entwicklung der Computerwelt. Zuerst gab es eine Aufteilung in zwei Gruppen, die der Homecomputer und die der Personalcomputer. In letzter Zeit ist ein deutlicher Trend in Richtung Atari zu sehen, und zwar vom 260 ST an aufwärts. Am 25.11.1986 erfolgte offiziell die Gründung der neuen Atari-Abteilung.

Das Clubleben besteht aus Treffen in loser Folge. Zu uns kann jeder kommen, der Rat und Hilfe braucht.

Ulrich Sauer
Danziger Straße 1
8754 Großostheim 2

Solingen

Der Solinger Atari-Club sucht noch Mitglieder in ganz Deutschland. Jedes Mitglied muß zwar einen Jahresbeitrag von 10.- DM bezahlen, kann aber die Vorzüge des Clubs genießen (z.B. die Clubzeitung).

Hans-Jürgen Müller
Bleichstraße 21
5650 Solingen 1

Bottrop

Der Atari ST User-Club Bottrop besteht erst seit einigen Wochen, hat aber schon 15 Mitglieder. Wir bieten eine Public-Domain-Software-Bibliothek, den Bezug von verbilligter Hard- und Software, eine vierteljährlich erscheinende Clubzeitung, Hilfe bei Soft- und Hardware-Problemen durch Experten und noch vieles mehr. Info gibt es gegen 80 Pf. Rückporto.

1. Atari ST User-Club Bottrop
Andreas Czech
In der Schanze 20
4250 Bottrop

Kiel

Unser Club (8 Bit) sucht Mitglieder aus dem ganzen Bundesgebiet. Wir tauschen, vermitteln und helfen unseren Mitgliedern nicht nur, sondern passen uns den Wünschen der Teilnehmer an. Es gibt auch keine Clubgebühren.

Lars Freymark
Masurenring 52
2300 Kiel 14

Bundesverband Deutscher Computerclubs

Der Computerclub Leonberg plant die Gründung einer Dachorganisation für die Computerclubs. Angestrebt wird hier der Zusammenschluß der vielen kleinen Vereinigungen, ohne deren Selbständigkeit zu beeinträchtigen.

Die Clubs sollen zu Landesverbänden zusammengefaßt und unter "Aufsicht" des Bundesverbandes gestellt werden. Ziel ist die Unterstützung der oft schlecht organisierten kleinen Clubs und die Koordination ihrer Aktivitäten. Ebenso könnte eine überregionale Clubzeitung erscheinen.

Der Computerclub Leonberg ruft alle Interessenten auf, sich zu melden und beim Aufbau dieser Organisation behilflich zu sein.

Computerclub Leonberg
Marcus Schindler
Uhlandstraße 15
7250 Leonberg 6
Tel. 07125/21822

Atari-XL/XE-Public-Domain-Softwareclub

Ich habe für Public-Domain-Programme einen Softwareclub gegründet, der als Anlaufstelle gedacht ist. Hier die Bedingungen zum Mitmachen:

- Es gibt keine Clubgebühr.
- Es dürfen nur Eigenentwicklungen eingeschickt werden.
- Dazu muß ein adressierter und frankierter Rückumschlag beiliegen.
- Wer eine Diskette einschickt, bekommt eine mit neuen Programmen zurück.
- Es sollte sich dabei um DOS 2.X handeln. Auch muß dabeistehen, ob die Floppy nur im 1040er-Modus arbeitet oder auch das 720er-Format verwendet werden kann.
- Turbo-Basic wäre günstig, muß aber nicht unbedingt sein.

André Cichowski
Pumpwerkstraße 17a
6800 Mannheim 1

Hallo Atari-User!

Ich suche Kontakt zu Atari-Usern in meiner Gegend (Atari 800XL mit Floppy oder Datasette).

Eric Michenfelder
Drosselweg 2
6839 Oberhausen



Pascal und Tennis

CompuCamp, Anbieter für Computer- und Sportferien, hat sein neues Programm vorgestellt. 1987 besteht in 6 Camps in Deutschland und Österreich die Möglichkeit, in den Ferien an einem Computer- oder Sportkurs oder auch an beidem teilzunehmen. Die Firma CompuCamp bietet 1987 für junge Urlauber 22 Computer- und 11 Sportkurse an.

Die seit 1983 ständig erweiterten Computerkurse über Basic, Pascal und Maschinensprache sowie die Spezialkurse wie Hardware-Basteln und DFÜ sollen Schülern und Jugendlichen den Einstieg in die Computerwelt erleichtern.

Erstmals in diesem Jahr werden auch spezielle Kurse für den Atari ST angeboten. GFA-Basic, die Programmiersprache C und die Benutzerschnittstelle GEM sind die Themen, mit denen sich Atari-ST-User in Westensee und Benediktbeuren auseinandersetzen können. Ebenfalls an diesen beiden Orten laufen Kurse für den Atari XL.

Was den Sport betrifft, so wird neben Tennis, Surfen und Segeln auch ein American-Sports-Kurs angeboten, bei dem man amerikanische Sportarten kennenlernen und selbst ausüben kann.

Die Kurse dauern jeweils eine Woche und können, da sie aufeinander aufbauen, auch über mehrere Wochen gebucht werden. Voraussetzung ist allerdings, daß man das nötige Kleingeld besitzt: Eine Woche im Computer- und Sportcamp kostet ab 595.– DM.

CompuCamp GmbH
Goßlerstraße 21
2000 Hamburg 55
Tel. 040/861255

Keine Chance für den Staub

Staubschutzhauben für sämtliche Atari-Computer bietet die Firma Rausch & Haub an. Sie sind für insgesamt 50 verschiedene Geräte ab Lager lieferbar. Sonderanfertigungen sind möglich. Auch für die Drucker von Star, Taxan, Okidata, NEC und Brother sowie für die wichtigsten sonstigen Peripheriegeräte läßt sich so das Staubproblem lösen.

Für Besitzer von Komplettanlagen werden Sets angeboten, deren Preis niedriger ausfällt als die Gesamtkosten für die Einzelstücke. Außerdem räumt die Firma Clubs, Vereinen und anderen Abnehmern von größeren Mengen Sonderrabatte ein.

Rausch & Haub
Berliner Freiheit 16
5300 Bonn 1
Tel. 0228/638313

Farbig digitalisieren

Der F.E.T. Farb-Videodigitizer ist das erste Gerät seiner Art, welches das Farbsignal einer Videokamera in 16 Echtfarben (Originalfarben) digitalisieren kann. Dazu benötigt er auf dem Atari 520 ST etwa 1,3 Sekunden. Somit können Videoaufnahmen sehr schnell digitalisiert werden. Diese Bilder lassen sich anschließend auf Diskette speichern und mit Grafikprogrammen wie "Degas" oder "Neochrome" weiterverarbeiten.

Die Einsatzmöglichkeiten eines solchen Digitizers sind vielfältig. Sie reichen von der Erstellung von Grafiken für Computerspiele und Animation bis hin zur technischen Anwendung.



Der Lieferumfang umfaßt den als Steckmodul ausgeführten Digitizer, die nötige Software und die Bedienungsanleitung. Der Preis beträgt für den Atari ST ca. 530 DM.

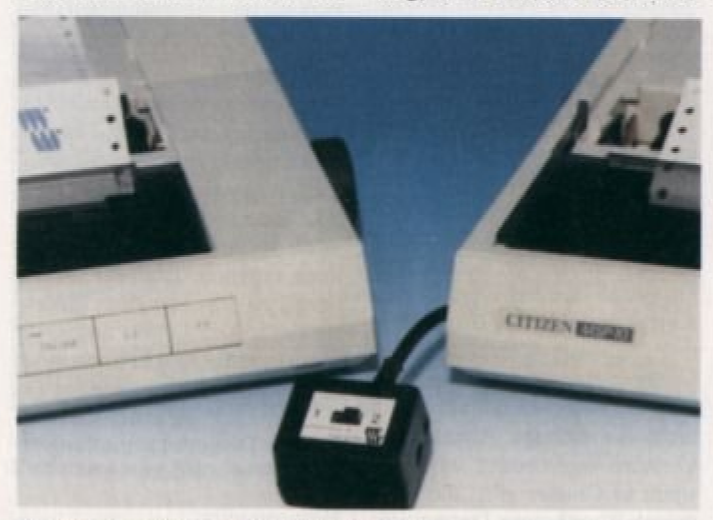
Creative Video
Postfach 1501
8520 Erlangen
Tel. 09195/2728

Schlaue Kabeleien

Die Firma Wiesemann hat sich im Bereich der Interfaces und Computerkabel bereits einen Namen gemacht. Mit intelligenten Lösungen werden die Probleme bei der Verbindung von Computer und Peripherie beseitigt. Jetzt hat Wiesemann eine Fülle von Produktneuheiten vorgestellt.

Die Verbindung von drei bidirektional arbeitenden V.24-Geräten erlaubt die V.24-T-Box, ein Kabel, an dessen einem Ende einfach zwei serielle Schnittstellen angeschlossen werden können. Die V.24-T-Box führt eine elektronische Oder-Verknüpfung zwischen zwei Slave-Anschlüssen und einem Master durch. Daten von jedem Slave gelangen an den Master, solche vom Master an beide Slaves. Für Anwendungen, bei denen sich gleichzeitiges Senden vermeiden läßt, bietet sie eine gute Lösung, da nicht umgeschaltet werden muß.

Für den Anschluß von zwei Druckern an einen Computer gibt es bereits viele Lösungen. Oft handelt es sich dabei um unhandliche und teure T-Switches, für die man erst noch drei Kabel zukaufen muß. Das Two-Printer-Cable bietet eine praktikablere Lösung zum günstigen Preis: Ein kleiner kompakter Umschalter ist einfach in den Kabelsatz integriert, so daß beide eine Einheit bilden. Vom Rechner geht ein 2 m langes Kabel zum ersten Drucker, von dort ein weiteres (ebenfalls 2 m lang) zum zweiten. Die kleine Bedienbox ist über ein einziges, 1 m langes Kabel nach vorn geführt, so daß der sonst übliche Kabelsalat aus dem Arbeitsbereich ferngehalten wird.



Praktische Lösung: Two-Printer-Cable

Eine komfortablere Lösung bietet der Multi-Printer-Adapter. Ein elektronischer Umschalter leitet die Daten an den Drucker, der manuell On-Line geschaltet wurde. Durch Kombination dieser Adapter lassen sich bis zu 16 Drucker an einen PC anschließen. Eine zusätzliche Stromversorgung ist nicht nötig. Dank CMOS-Bausteinen genügen die Signalspannungen.

Daten der seriellen V.24-Schnittstelle lassen sich nur über kleine Entfernungen übertragen. Abhilfe schafft der RS-232-Booster. Dieser Datenantreiber beschleunigt die Übertragungsgeschwindigkeit von 20 KBit/s auf 100 KBit/s und erlaubt Entfernungen bis 1,2 km.

Wiesemann
Winckenbachstr. 3-5
5600 Wuppertal
Tel. 02 02 / 50 50 77

EDV-Berufe

Schätzungsweise 30.000 bis 40.000 offene Stellen gibt es im Bereich der EDV. Der ständig noch wachsende Bedarf an EDV-Fachleuten kann durch die Berufsanfänger nicht befriedigt werden.

Diesem Bereich der Aus- und Fortbildung widmet sich neben anderen Ausbildungsstätten auch das Control Data Institut, ein herstellerunabhängiges Unternehmen, nach eigenen Angaben das größte Europas. Es hat soeben seinen neuen Katalog für 1987 herausgebracht, der das Angebot für die Ausbildung in EDV-Berufen enthält und die Vermittlung von Computerqualifikation bietet. Die Palette reicht vom bekannten Computerführerschein bis zum Organisationsprogrammierer.

Control Data Institut
Karlstraße 42
8000 München 2
Tel. 0 89 / 52 39 10



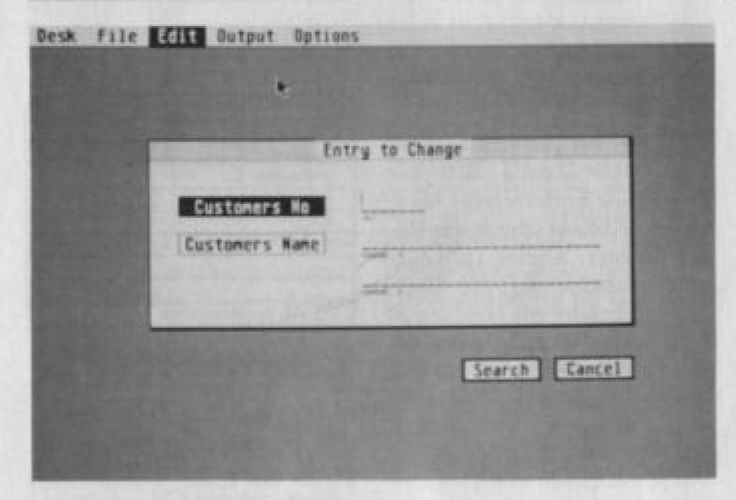
Etiketten für 3,5″

Diese Etiketten für 3,5″-Disketten machen dem üblichen Chaos ein Ende. Durch einen speziellen Aufdruck erleichtern sie eine systematische und sinnvolle Registrierung aller Disketten. Geliefert werden die Etiketten in Packungen mit 48 Stück in vier verschiedenen Farben. Sie sind im einschlägigen Fachhandel erhältlich oder direkt vom Hersteller zu beziehen.

Technik & Design
Postfach 1170
8550 Forchheim/Ofr.
Tel. 0 91 91 / 1 40 49

Software-News

Zur Freude aller ST-Besitzer sind die anfänglichen Software-Probleme weitgehend beseitigt. Zur Zeit gibt es eine regelrechte Schwemme guter Programme, was besonders für den Anwenderbereich gilt. Da wir im **ATARI***magazin* umfassend berichten wollen, aus Platzgründen aber nicht auf jedes neue Programm ausführlich eingehen können, werden wir an dieser Stelle einige nur in Kurzform vorstellen. Dabei kann es durchaus vorkommen, daß wir über das eine oder andere zu einem späteren Zeitpunkt noch mehr berichten werden. Jetzt aber Ring frei zur ersten Runde.



Zugangssicherung: AS-Adress

AS Adress

Bei diesem Programm handelt es sich, wie der Name schon sagt, um eine Adreßverwaltung mit eigener Druckausgabe. Neben komfortablen Such- und Druckoptionen und der Möglichkeit der Serienbrieferstellung bietet "AS Adress" den Besitzern einer Harddisk (Festplatte) eine besondere Hilfestellung. Außer zwei Floppy-Laufwerken kann der Anwender auch vier Harddisk-Partitions ansprechen. Außerdem ist eine komplette Backup-Einrichtung installiert.

Insgesamt gesehen bietet "AS Adress" damit mehr als vergleichbare Programme und kann deshalb empfohlen werden, wenn man sich nicht an den englischen Programmkommentaren stört (die Anleitung ist deutsch).

Hersteller/Bezugsquelle:
Application Service, Herne

PKS Mail

So nennt sich ein Adreßverwaltungsprogramm, das sowohl als Desk-Accessory wie auch als normales Programm gestartet werden kann. Zwei Punkte heben es von der Masse ab. Der erste ist die einfache Bedienung, die allerdings zur Folge hat, daß man sich hier auf eine Eingabemaske festlegt, in welche die gängigen Daten einer Adressendatei eingetragen werden können. Auch kann man mit "PKS Mail" alleine keine Daten ausdrucken.

Damit wären wir auch gleich beim zweiten Punkt. Durch die Gestaltung als Accessory kann man z.B. erst "1 ST Word" laden, einen Brief schreiben und dann "PKS Mail" aufrufen, um etwa Serienbriefe zu erstellen. Auch die Anfertigung von Formularen wie z.B. Schecks, Zahlkarten und Postaufkleber wird unterstützt.

Für größere Datenmengen ist "PKS Mail" allerdings nicht geeignet, obwohl rein rechnerisch je nach Speicherkapazität und Datensatzlänge zwischen 5000 bis 10.000 Adressen verwaltet werden können. Für eine solche Menge müßte das Programm dann aber doch einige zusätzliche Features bieten.

Hersteller/Bezugsquelle:
Pahlen & Krauß

Diskmenü für den ST

Mit "Diskmenü" kann eine beliebige Anzahl von Dateien in beliebiger Größe von der Harddisk auf Disketten gesichert werden. Die Dateien werden dabei gepackt hintereinander in eine Datei geschrieben. Dateien, die zu groß für eine Diskette sind, werden auf mehrere Disketten verteilt. Die gesicherten Dateien können einzeln oder zusammen wieder zurückkopiert werden.

Doch "Diskmenü" kann mehr, als nur Dateien sichern. Das Programm ist auch in der Lage, die Harddisk auf Fehler hin zu überprüfen. So überprüft es das Inhaltsverzeichnis der Diskette/Harddisk auf seine Korrektheit hin und ob alle Dateien richtig im Verzeichnis eingetragen sind. So wird zum Beispiel erkannt, wenn einer Datei nicht die richtige Anzahl von Clustern zugeordnet ist, oder wenn es Cluster gibt, die zwar belegt, aber keiner Datei zugeordnet sind.

"Diskmenü" druckt auf Wunsch eine Liste mit den Angaben darüber aus, welcher Cluster von welcher Datei belegt wird oder welche Datei welche Cluster belegt. So erhält man eine ausführliche Inhaltsliste der Harddisk / Diskette. Da "Diskmenü" ganz unter GEM arbeitet, ist es sehr komfortabel und leicht zu bedienen. Das Programm kostet 98.– DM und wird mit einer deutschen Anleitung geliefert.

Computerware Gerd Sender
Moselstraße 39
5000 Köln 50

Katpro Royal

Wer langsam den Überblick verliert, was auf welcher Diskette gespeichert ist, sollte sich näher mit "Katpro Royal" beschäftigen. Dieses Diskettenverwaltungsprogramm soll dem Anwender helfen, Ordnung zu wahren.

Je nach Speicherkapazität können bis zu 20.000 Dateien und Ordner verwaltet werden. Sie lassen sich nach Name, Größe, Extension usw. sortieren oder suchen. Natürlich ist es auch möglich, Einträge zu ändern, zu löschen oder auszudrucken. Die Datei- und Ordnerinformationen können automatisch eingelesen werden, was die Bearbeitung einer umfangreichen Diskettensammlung erheblich erleichtert.

Bezugsquelle:
Boston Computer



Ordnung auf den Disketten: Katpro Royal

C-Soft

Unter diesem Label werden einige Business-Programme gehobenen Niveaus angeboten, von denen uns folgende als Demoversion vorlagen (diese Demos können beim Hersteller bestellt werden):

"C-Auftrag": Das Programm beinhaltet Kunden-, Lieferanten-, Artikel/Lager- und Auftragsverwaltung. Außerdem wird das Textverarbeitungsprogramm "C-Text Plus" mitgeliefert. Das Paket arbeitet mit der Finanzbuchhaltung aus gleichem Hause zusammen.

"C-Fibu": Hier sind Debitoren-, Kreditoren- und Sachkonten möglich. Außerdem können Daten aus der Rechnungsschreibung von "C-Auftrag" und offene Posten mit Mahnwesen übernommen werden.

"C-Adress" bietet eine komfortable Adreßverwaltung und ermöglicht in Zusammenarbeit mit "C-Text" die Erstellung von Serienbriefen usw.

"C-Dent": Hier handelt es sich um spezielle Software für Zahnärzte. Das Programm betrifft den Prothetikbereich mit Zahnschema, Kiefermodell sowie Heil- und Kostenplan. Außerdem ist ein Statistikteil integriert.

Weitere Informationen:

C-Soft
Holzfällerstr. 4
8400 Regensburg



Umfangreiche Kundendaten: C-Auftrag

Simulation: Cardiac Arrest

Für alle Atari 800/130/ST-User, die sich beruflich oder privat für Notfallmedizin interessieren, liegt jetzt in den USA ein geeignetes Trainingsprogramm vor. "Cardiac Arrest" simuliert 45 Patienten, deren Leben es zu retten gilt. Der Computer zeigt die Krankengeschichte sowie ein EKG. Der Anwender kann nun diverse Notfallmedikamente verabreichen, defibrillieren und verschiedene weitere Therapieformen einsetzen. Auch eine Blutgasanalyse steht zur Verfügung. Nach Abschluß der mehr oder weniger erfolgreichen Behandlung erfolgt eine Bewertung des Vorgehens.

"Cardiac Arrest" beinhaltet eine ausführliche Dokumentation, welche nicht nur die EKG-Bilder sowie Wirkung und Dosierung der Medikamente erläutert, sondern außerdem konkrete Therapievorschläge für die einzelnen Fallbeispiele bringt.

Die Adresse der Softwarefirma, die auch andere Programme aus dem medizinischen Bereich vertreibt, lautet:

Mad Scientist Software
2063 North 820 West
Pleasant Grove, UT 84062
USA



Neue Fibu

Das Programm "BS-Fibu", eine mandanten- und mehrfirmenfähige Finanzbuchhaltung der Firma Bavaria-Soft, ist jetzt lieferbar. Es wurde für den professionellen Einsatz auf dem Atari ST entworfen und bietet unter anderem folgende Leistungsmerkmale:

- mehrfirmenfähig
- Kreditoren-/Debitorenverwaltung
- frei wählbarer Kontenrahmen
- offene Posten
- Umsatzstatistik
- Umsatzsteuervoranmeldung
- Gewinn- und Verlustrechnung
- betriebswirtschaftliche Auswertung
- Summen- und Saldenlisten

Geliefert wird das Programm mit einem 100seitigen Handbuch.

Bavaria-Soft Datentechnik GmbH
Otto-Hahn-Straße 25
8012 Ottobrunn bei München
Tel. 089/6097838

Jack Paint

Hinter dieser Bezeichnung verbirgt sich – wie zu erwarten – ein Zeichenprogramm für Schwarzweiß-Monitore. Trotz des relativ geringen Preises bietet es eine Menge. Auffällig ist der Verzicht auf die Pull-Down-Menüs, mit denen z.B. "Monostar" arbeitet. Bei "Jack Paint" stehen alle Optionen in Symbolform (mit Bildunterschriften) nach dem Laden direkt auf dem Bildschirm. Aus dieser Gesamtübersicht, zu der man nach jedem Arbeitsgang zurückkehrt, kann man per Mausklick auswählen. Insgesamt lassen sich zwei Bilder im Speicher bearbeiten.

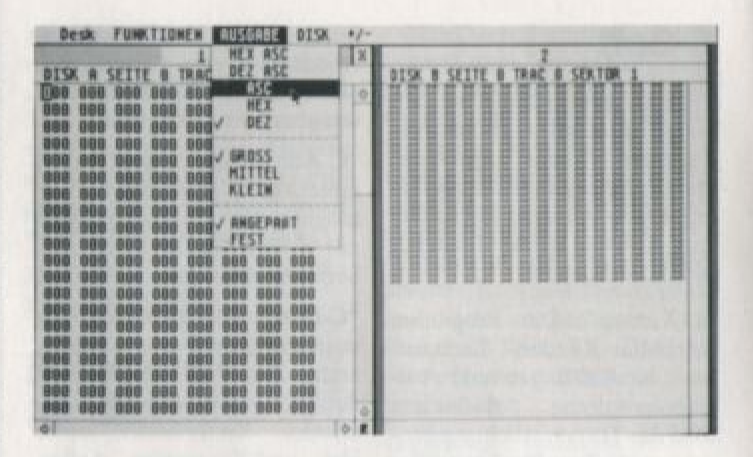
"Jack Paint" erlaubt alle Operationen, die man auch von anderen Programmen dieser Art kennt, z.B. Freihandzeichnen, geometrische Formen, Lupenfunktion usw. Auf Besonderheiten wie Dehnen und Biegen wurde verzichtet. Aus diesem Grund halte ich "Monostar" für das bessere Programm. Wer aber solche Sonderfunktionen nicht benötigt, sollte sich "Jack Paint" einmal ansehen.

Hersteller: R & P Soft, Cham

G Diskmon

Dieser Diskettenmonitor nutzt die unter GEM angebotenen Möglichkeiten voll aus und ist damit sehr anwenderfreundlich. Er bietet alle Features, die man von einem solchen Programm erwartet.

Man kann den Disketteninhalt auf den Bildschirm bringen (2 Fenster zeigen die Vorder- und die Rückseite der Diskette gleichzeitig an), ihn ausdrucken, schreiben, lesen, kopieren, einzelne Tracks formatieren, Strings suchen usw. Infolge der Pull-Down-Menüs wird die deutsche Anleitung nur selten gebraucht. "G Diskmon" stellt ein solides Werkzeug dar, das für alle, die mit Disketten arbeiten, unerläßlich ist.



Blick auf beide Diskettenseiten: G-Diskmon

Bezugsquelle: G Data

K-Graph 2 und K-Spread 2

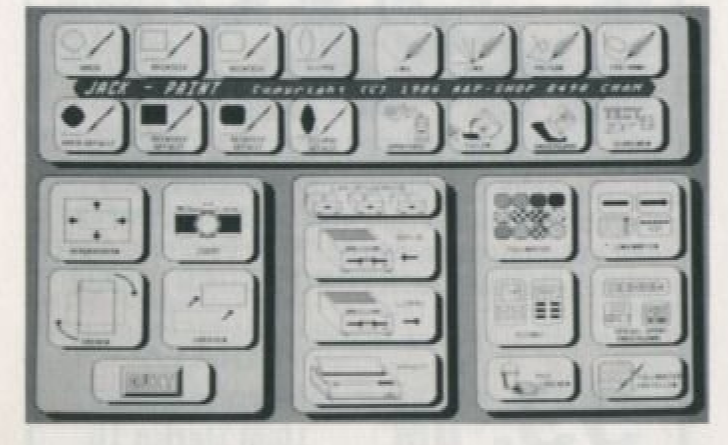
Anwendern des ST sind die Produkte von Kuma ein Begriff. Zwei Programme dieser Software-Firma werden jetzt in verbesserter Version ausgeliefert.

"K-Spread 2" vereinigt in sich eine beachtliche Fülle von Möglichkeiten, die voll in GEM mit seinen Fenstern und der Maussteuerung integriert sind. Hier eine kurze Auflistung:

- 256 Spalten bei 8126 Zeilen
- gleichzeitig bis zu 5 Fenster
- über 60 Funktionen
- Makros
- alphanumerisches Sortieren
- weitgehende Formatierung der Bildschirmausgabe
- Querausdruck auf Epson-kompatiblen Druckern
- Tastenkombinationen als Alternative zu den Pull-Down-Menüs
- direkter Zugriff auf den Drucker aus dem Rechenblatt heraus

"K-Spread 2" kann alleine oder in Kombination mit anderen Kuma-Produkten angewandt werden. Eines davon ist "K-Graph", das nun ebenfalls in einer neuen Version vorgestellt wurde. Dabei handelt es sich um ein Programm für Geschäftsgrafiken, das es dem Anwender ermöglicht, schnell Balken- und Tortengrafiken von seinen Daten anzufertigen. Natürlich läuft auch "K-Graph" voll unter GEM und ist damit sehr benutzerfreundlich.

Die neuen Möglichkeiten von "K-Graph 2" bestehen u.a. in mathematischen und trigonometrischen sowie umfangreichen Statistikfunktionen.



Optionen auf einen Blick: Jack Paint

Neues aus USA

Wer eine Reise durch unser Sonnensystem unternehmen oder mehr über die Planeten und ihre Monde erfahren will, sollte sich "The Halley-Project" näher ansehen.

The Halley-Project

Ziel des Spiels ist es , vom Kometen Halley aus verschiedene Missionen in möglichst kurzer Zeit zu erfüllen. Die ersten Aufgaben, unter anderem ein Flug zur Erde, machen wenig Schwierigkeiten. Wenn man aber in höheren Stufen einen Mond finden muß, der wärmer als Callisto ist, kommt man wohl nicht umhin, diverse Nachschlagewerke zu Rate zu ziehen.

Die Orientierung im Weltraum erfolgt anhand der Sternbilder, der Sonne und des Radar-Screens unseres Weltraumvehikels. Die Entfernungen zwischen den Himmelskörpern stimmen in dieser Simulation auf den Kilometer genau. Um den Spieler aber nicht jahrelang durch das All fliegen zu lassen, tritt das Raumschiff allen physikalischen Gesetzen zum Trotz mit dem Erreichen der Lichtgeschwindigkeit in den Hyperraum ein, der es schneller von Planet zu Planet gelangen läßt. Dennoch bekommt man durch die Dauer dieser Sequenz einen Eindruck von der unterschiedlichen Distanz zwischen Erde und Mars bzw. Erde und Pluto.

Der Anflug zum Ziel ist ein grafisches Meisterwerk. Die Monde kreisen beim Näherkommen um den Planeten, und die der Sonne abgewandte Seite liegt wie in Wirklichkeit im Schatten. Schwierig gestaltet sich die Suche nach dem Landepunkt am Zielort, und nicht selten handelt man sich durch die Kollision mit einem Himmelskörper Strafminuten ein. Nach sicherer Landung wird der Spieler mit einem Panorama des Planeten belohnt. Dann erhält er seine nächste Aufgabe.

Die Ausstattung des Spiels ist vorbildlich. Die orginelle Verpackung bietet neben der Diskette (Atari/Commodore-Floppy) eine ausführliche Anleitung, eine Referenzkarte mit den wichtigsten Steuerungsfunktionen, eine Sternkarte und als Gimmick eine Cassette, die den Piloten instruiert.

"The Halley-Project" ist ein gut gemachtes Programm, das den Spieler lange fesselt und bei dem neben dem Vergnügen auch das Lernen nicht zu kurz kommt.

Hundstage in Fooblitzky

Infocom hat ein neues Spiel herausgebracht. "Fooblitzky" (Autor: Michael "Infidel" Berlyn) ist eine Premiere für das erfolgreiche Softwarehaus. Dieses Produkt besitzt Grafiken (!) und stellt ausnahmsweise kein Adventure, sondern ein kniffliges Brettspiel für vier Personen dar. Die Ausstattung entspricht dem gewohnten Infocom-Standard. Neben der Diskette finden sich in der Packung vier aufklappbare, abwaschbare Tafeln, jede mit einem Stadtplan, Checkliste und Raum für Notizen versehen, vier Folienstifte sowie zwei Anleitungen, eine ausführliche für Anfänger und eine stichpunktartige für Fortgeschrittene.

Worum geht es nun bei "Fooblitzky"? Jeder Spieler schlüpft in die Rolle eines Hundes, der auf seinen Streifzügen durch die Stadt in diversen Läden (Spielwarengeschäft, Drogerie usw.) vier Dinge erwerben und zu einem Checkpoint bringen muß. Das klingt leichter, als es ist. Von den vier Gegenständen kennt jeder Spieler nur einen einzigen, den er zu Beginn heimlich ausgewählt hat. Die restlichen drei können nur durch genaues Beobachten der Kaufgewohnheiten der Mitstreiter und durch logisches Denken (am Checkpoint gibt der Computer à la Mastermind die Zahl der richtigen und falschen Waren an) ermittelt werden.



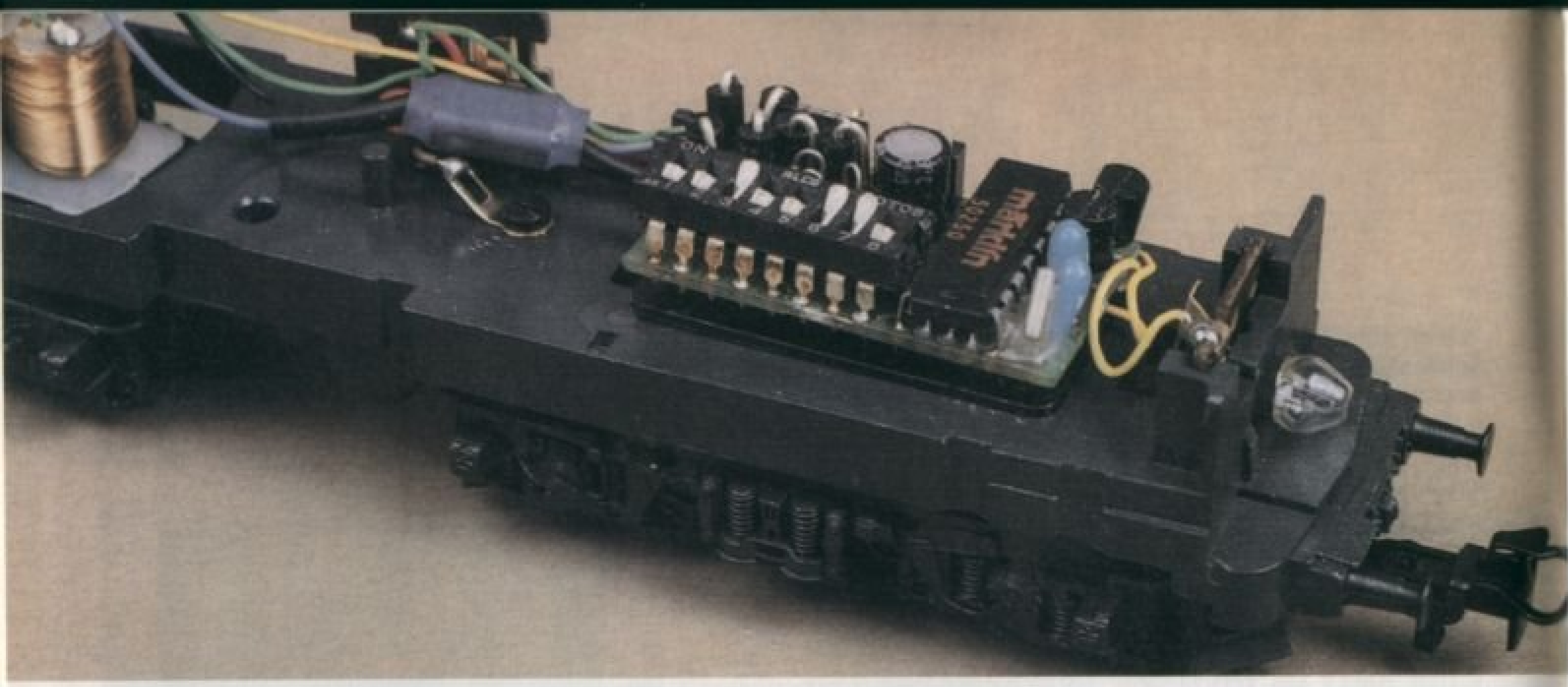
The "Halley-Project" erschließt die Dimensionen des Alls. Der Spieler soll sich zwischen den Sternen orientieren.

Das Spiel wird ausschließlich mit dem Joystick gesteuert. Die ausgezeichnete Grafik ist in Graphics 8 gehalten und erscheint daher auf europäischen PAL-Fernsehern leider schwarzweiß. Im Spielverlauf tauchen Windows auf dem Stadtplan auf, die mit vielen Cartoons die Handlungen der Spieler illustrieren. Die Animation ist nicht ganz flackerfrei, da das Programm von der Player-Missile-Grafik des Atari keinen Gebrauch macht. Wahrscheinlich soll so die Übersetzung auf andere Systeme erleichtert werden. Etwas nervend ist der ständige Zugriff auf Diskette. Darüber tröstet aber die Tatsache hinweg, daß das Programm auf anderen Rechnern nur mit 128 KByte läuft.

Frank Emmert



In "Fooblitzky" können Sie leicht zum armen Hund werden. In der Rolle des Vierbeiners müssen Sie heil durch die Stadt kommen, um verschiedene Gegenstände zu erwerben.



Mit dem ST am Zug

"Digital" heißt die neue Dimension bei den Modelleisenbahnen. Mit Computeranschluß kommt die Sache richtig in Gang. Wir zeigen Ihnen wie!

Während so mancher Sprößling seine Freizeit mit Bits und Bytes zwischen Computer, Floppy und anderen Fremdwörtern verbringt, vergnügt sich mancher Vater mit einer elektrischen Modelleisenbahn.

Von Märklin, dem wohl bekanntesten Modelleisenbahnhersteller in Deutschland, kommt nun eine Technik, die die beiden Generationen wieder an einen Tisch bringen könnte: Märklin Digital – die Modellbahn mit Computeranschluß.

Eigentlich ist der Computeranschluß bei diesem elektronischen Mehrzugsteuersystem nur ein Nebenprodukt; auch ohne Rechner leistet Märklin Digital Erstaunliches. Auf einem Gleis können bis zu 80 Lokomotiven und bis zu 256 Magnetartikel (Weichen, Signale, Entkupplungsgleise usw.) gesteuert werden – über nur zwei Kabel!

Bisher kann das System nur mit der Märklin-H0-Eisenbahn voll ausgeschöpft werden (Maßstab 1:87 und 3-Leiter -Wechselstrom-System). Die Magnetartikel-(Weichen-)steuerung läßt sich für alle gebräuchlichen Modellbahnen verwenden, unabhängig von Maßstab und Hersteller. Wenn ab 1987 besondere Bausteine für Gleichstrommotoren ausgeliefert werden, können auch Lokomotiven vieler anderer Hersteller digitalisiert werden.

Herzstück von Märklin Digital ist die Central-Unit, ein unscheinbares Kästchen, das durch zwei Kabel mit dem Gleis verbunden wird. Die Stromversorgung wird von einem eigenen Trafoteil übernommen (außer dem 50 VA starken Digital-Transformer kann auch ein normaler 16V-Wechselstromtrafo verwendet werden). Da an den Gleisen immer volle Spannung anliegt, erhält man als Nebeneffekt eine konstante, gleichmäßige Innenbeleuchtung von Loks und Wagen. Auch die Magnetartikel können den benötigten Strom zum Schalten dem Gleis entnehmen.

Rechts und links an dieser Zentraleinheit lassen sich nun verschiedene weitere Geräte anstecken, was über eine verpolsichere Steckerleiste geschieht. An der linken Seite ist der Anschluß der Keyboards möglich, auf denen sich 16 Paare aus je einer roten und einer grünen Taste befinden. Mit solch einem Pult können dann 16 Doppelspulenantriebe, wie sie in Weichen und Signalen enthalten sind, oder die doppelte Menge an (einspuligen) Entkupplungsgleisen (bzw. Kombinationen davon) gesteuert werden. Da sich bis zu 16 Keyboards anschließen lassen, sind bis zu 256 Weichen (Doppelspulenantriebe) möglich. Das Stellen der Weichen hat sich gegenüber dem gewohnten Stellpunkt 7072 nicht verändert, nur daß beim digitalen Keyboard die Stellung RUND (Rote Taste) durch eine rote LED angezeigt wird.

Ebenfalls links können auch die neuen Memory-Geräte angereiht werden. Damit lassen sich bis zu 24 Fahrstraßen speichern, die dann mit einem einzigen Knopfdruck aufgerufen werden können. Auf Wunsch kann auch ein Profi-Modus eingestellt werden, der dann z.B. kreuzende Fahrstraßen nicht zuläßt. Auch Schattenbahnhof- und Blockstellensteuerung lassen sich mit dem Memory verwirklichen.

An die rechte Seite der Central-Unit kommt das Control 80, ein Gerät, das aufgrund seines Drehreglers oft mit normalen Trafos verwechselt wird. Zusätzlich sind jedoch eine Zehnertastatur, eine zweistellige 7-Segment-Anzeige und die Tasten STOP, GO, FUNKTION und OFF zu sehen.

Bevor eine Lok mit dem Drehregler angesprochen werden

kann, muß deren Nummer mit der Zehnertastatur eingestellt werden; sie erscheint dann in der Leuchtanzeige. Welche Nummer eine Lokomotive erhalten soll, kann der Lokführer am Decoder selbst bestimmen. Ist die (immer zweistellige) Nummer aktiviert, so kann die gewählte Lok mit dem Drehknopf wie gewohnt gesteuert werden. Auch das Umschalten der Fahrtrichtung durch Drehen des Reglers über die Nullstellung hinaus wurde vom konventionellen Trafo übernommen.

Mit der Taste FUNKTION kann jederzeit eine Funktion an der Lok betätigt werden (in der Regel das Licht, das nun fahrtrichtungsabhängig brennt, d.h. immer an der Frontseite). Stehen andere Einrichtungen zur Verfügung, so können diese stattdessen ferngesteuert werden (z.B. Telexkupplung oder Rauchgenerator).

Hat man nun eine Lok auf die Reise geschickt und gibt dann eine neue Nummer ein, so setzt sie ihre Fahrt mit der zuletzt eingestellten Geschwindigkeit und dem Funktionsstatus fort, bis sie neue Befehle erhält. Gleichzeitig reagiert sie jedoch auf konventionelle Steuerungsmethoden wie Signale mit Zugbeeinflussung, die bei Rot die Stromversorgung für einen bestimmten Gleisabschnitt einfach unterbrechen. Allerdings sollten für digitalisierte Anlagen die Trennstellen mit einem kleinen Widerstand überbrückt werden, da der Empfängerbaustein in der Lok sonst nach ca. 2 bis 4 Minuten seinen Auftrag vergißt.

So können nun nacheinander bis zu 80 Lokomotiven auf die Reise geschickt werden – theoretisch zumindest, da auch dem relativ starken Digital-Transformer bei mehr als 4 bis 5 gleichzeitig fahrenden Loks die Luft ausgeht. Wenn mehrere Loks zur gleichen Zeit fahren sollen, muß zudem für jeden Wechsel zwischen den Maschinen die Nummer wieder eingetippt werden. Daher wird wohl jeder Modellbahner früher oder später ein weiteres Control 80 anstecken. Bis zu 10 dieser Geräte lassen sich gleichzeitig anschließen, wobei eine Loknummer selbstverständlich nur auf einem Gerät eingestellt sein kann. Versucht man es auf einem anderen, so blinkt die Zahl als ungültig auf.

Erscheint einmal ein Zusammenstoß unvermeidlich, kann mit der STOP-Taste von jedem Control 80 aus die Notbremse gezogen werden. Die Stromzufuhr der Anlage wird dann unterbrochen, und man hat genügend Zeit, den Knoten wieder zu entflechten. Dieser Notstopp, der auch durch einen (versehentlichen) Kurzschluß ausgelöst wird, kann mit der Taste GO von jedem Control 80 wieder freigegeben werden.

Nun jedoch zum fahrenden Material, den digitalen Lokomotiven. Um aus einer konventionellen Lok ein solches vollelektronisches Wunderwerk zu machen, wird lediglich ein kleiner, unscheinbarer Baustein aus dem Hause Märklin eingebaut, der den konventionellen, mechanischen Umschalter ersetzt. Jetzt entfallen endlich der lästige Hüpfer und das Aufblitzen der Lichter beim Richtungswechsel. Das Geheimnis dieser Platine ist ein CMOS-IC, das die auf die Fahrspannung aufmodulierten Signale entschlüsselt und weitergibt. Da CMOS-ICs sehr empfindlich gegenüber statischer Elektrizität sind und der Einbau ein spezielles Meßgerät erfordert, muß dieser leider vom Händler ausgeführt werden. Für eigene Versuche übernimmt Märklin keine Garantieleistungen!

Da der Decoder in der Lage ist, die ankommende Fahrspannung zu analysieren, kann er sich auch auf den normalen Betrieb einstellen und auf einer konventionellen Anlage betrieben werden. So ist es auch möglich, zuerst einmal nur einen Teilbereich einer bereits vorhandenen Anlage zu digitalisieren.

Auch die Weichen werden von Decodern versorgt; jeweils vier Magnetartikel lassen sich an einem Weichendecoder K 83 anschließen. Er kann seine Informationen und die Schaltspannung entweder direkt vom Gleis oder einer zweiadrigen Ringleitung beziehen.

Alle Decoder enthalten eine Reihe DIL-Schalter, wie man sie häufig bei Druckern vorfindet. Über eine beigelegte Tabelle

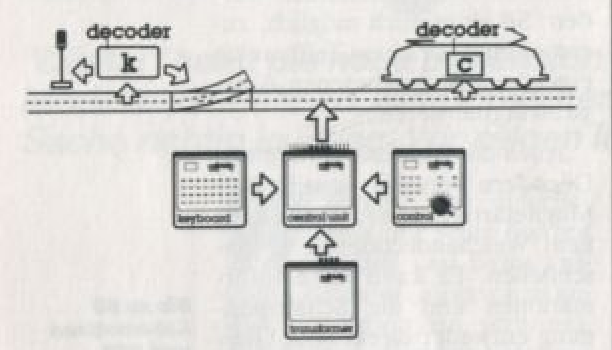
Bis zu 80 Lokomotiven und 256 Weichen können mit dem Digital-System gleichzeitig und unabhängig voneinander geschaltet werden.



kann die Nummer (bzw. Nummerngruppe bei Weichen) eingestellt werden, auf die der Baustein reagieren soll. Bei den Loks haben sich als Anhaltspunkt die ersten beiden Ziffern der Baureihe durchgesetzt (die bekannte V 200 fährt z.B. meist als 20).

Das Interface, das nun den Computer ins Spiel bringt, wird, wie die Fahrgeräte, ebenfalls auf der rechten Seite der Central-Unit angesteckt. Da es jedoch auf der rechten Seite nur eine sechspolige DIN-Buchse enthält, ist das Anstecken weiterer Geräte am Interface nicht möglich. Für die Buchse liegt dem Interface ein passender Stecker bei, für die Kabelverbindung zum Computer muß der Anwender dann selbst sorgen. Da lediglich 4 der 6 Pins belegt sind, ist dies aber recht einfach. Wie das Kabel für einen Atari ST aussehen muß, ist aus dem Schaltplan ersichtlich.

Transformator, Zentraleinheit, Fahrgerät und Stellpult sind für den Einstieg in's Digital-Zeitalter erforderlich.



Pin 1 ist die Datenleitung vom Computer zum Interface (RD).

Pin 3 ist die gemeinsame Masseleitung (Signal Ground).

Pin 4 ist die Datenleitung vom Interface zum Computer (TD).

Pin 5 ist die Steuerleitung für die Empfangsbereitschaft des Interface (CTS).

Über vier DIL-Schalter an der Rückseite des Interface-Bausteins läßt sich das Digital-System weitgehend an den jeweiligen Computer anpassen:

Schalter	Funktion on/off
1	RD negative/positive Logik
2	TD negative/positive Logik
3	CTS negative/positive Logik
4	GND TTL-Pegel (0/5 V) / RS-232-Pegel

Für den Atari ST müssen die Schalter 1 und 2 auf ON und 3 und 4 auf OFF gestellt werden.

Wenn die Verbindung zwischen Computer und Interface realisiert ist und alle Parameter richtig eingestellt sind, kann mit dem digitalen, computergesteuerten Fahrbetrieb begonnen werden. Drei Gruppen von Befehlen sind zu unterscheiden: Schaltbefehle, Lokbefehle und Nothalt (STOP/GO).

Am einfachsten sind STOP und GO, da sie jeweils aus einem Byte bestehen. Es genügt, den Code des jeweiligen Befehls ($61 für STOP und $60 für GO) auszugeben. Im Programm ist nur darauf zu achten, daß, während STOP aktiv ist, keine Befehle mehr ausgegeben werden, da die Anlage sonst abstürzen kann.

Um eine Weiche zu stellen, wird zuerst die Richtung (33 für gerade bzw. 34 für rund) und dann die Nummer der Weiche ausgegeben. Der Weichenantrieb muß dann mit dem Befehl 32 wieder abgestellt werden. Damit lassen sich auch schwerfälligere Antriebe oder andere Aufgaben bewältigen.

Um nun eine Lokomotive ins Spiel zu bringen, sind einige weitere Dinge zu berücksichtigen. Zunächst gibt man die gewünschte Fahrstufe aus, die von 0 bis 14 reichen darf. Soll die Funktion angeschaltet sein, so ist zu diesem Wert 16 zu addieren. Für den Fahrtrichtungswechsel ist der Wert 15 zu senden. Es ist bei der Computersteuerung also möglich, aus voller Fahrt heraus umzukehren – ob man dies allerdings dem Lokgetriebe zumuten sollte? Als zweiter Wert ist jetzt noch die Nummer der gewünschten Lok auszugeben – fertig!

Über den Rückmeldebaustein S 88 können Schaltgleise abgefragt werden, die beim Überfahren bestimmte Funktionen auslösen. Ist einer der 16 Eingänge des S 88 gegen Masse geschaltet, wird diese Tatsache so lange im Decoder gespeichert, bis der Status wieder vom Computer ausgelesen wird. So lassen sich mit dem Computer in gewissem Maße automatische Steuerungen übernehmen.

Leider ist die Kommunikation zwischen Computer und Central-Unit sonst aber recht spärlich. Es besteht keine Möglichkeit, den Status von Weichen oder Loks zu erfragen; nicht einmal über den STOP-Status ist der Computer informiert. Es wird zwar die Weichenstellung (LED) auf dem Keyboard akualisiert, wenn der Computer eine Weiche stellt, er kann im umgekehrten Fall jedoch nicht feststellen, wann eine Weiche direkt vom Keyboard oder einem anderen Gerät betätigt wird. Auch wenn eine Lok bereits auf einem Fahrgerät eingestellt ist, der Computer also im Moment nicht darauf zugreifen kann, ist dies vom Programm nicht abzufangen.

Wofür kann man nun den Computer sinnvoll einsetzen? In der Regel wird man seinen Rechner wohl hauptsächlich für die Weichensteuerung verwenden, denn bisher bietet der Computer die einzige Möglichkeit, ein Gleisbildstellwerk zu realisieren. Für die Steuerung von Zügen ist er nur in sehr geringem Umfang geeignet. Da er keine Möglichkeit hat, festzustellen, wo sich welche Lok zu welchem Zeitpunkt gerade befindet, ist ein programmgesteuerter Fahrplanbetrieb nur unter größtem Aufwand denkbar.

Ein Computer könnte zwar (im Interrupt) einen Wendezugbetrieb auf einer Nebenstrecke steuern (auf der dann nur eine Lok verkehrt, so daß klar ist, welche Lok den Umschaltimpuls erhalten soll), aber auch das ausgeklügeltste Programm kann mit einer entgleisten Lok oder ähnlichen Ereignissen nur sehr wenig

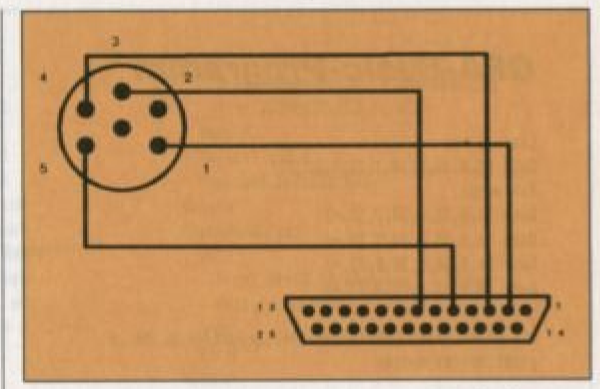
anfangen. Andererseits soll der Computer aber auch nur unterstützen und nicht den Modellbahn-Lokführer vom Trafo wegrationalisieren. Überhaupt ist das ganze System nicht dazu gedacht, dem Modellbahner Arbeit abzunehmen. Im Gegenteil, durch die vervielfachten Spiel- und Steuermöglichkeiten muß man oft ganz schön auf der Hut sein, daß der Intercity nicht plötzlich mitten durch den Rangierbahnhof prescht.

Nun aber genug der Theorie. Wir haben ein kleines Programm abgedruckt, das einige der Möglichkeiten von Märklin Digital nutzt. Es simuliert gleichzeitig fünf Fahrgeräte sowie Schalter für 10 Weichen und vier Fahrstraßen.

Das Programm wurde mit GFA-Basic erstellt und ist für einen S/W-Monitor ausgelegt. Die Bedienung sollte so komfortabel wie möglich programmiert werden, d.h., alle Funktionen lassen sich mit der Maus steuern. Die Geschwindigkeit der Loks wird über fünf Schieberegler eingestellt, die vertikal bewegt werden können (ähnlich den Slidern bei den Windows). Darüber ist die Taste F für die jeweilige Funktion angeordnet. Unter dem Regler befindet sich Taste U für den Richtungswechsel. Eine aktive Funktion wird durch eine inverse F-Taste angezeigt; darunter ist die aktuelle Fahrstufe abzulesen. Soll eine neue Nummer eingegeben werden, so ist auf die über der F-Taste befindliche Zahl zu klicken und diese durch Drücken der linken bzw. rechten Maustaste zu erniedrigen bzw. zu erhöhen. Ein gemeinsamer Druck auf beide Tasten verläßt diese Funktion wieder.

Auf der rechten Seite befindet sich am unteren Rand das Stellpult für 10 Magnetartikel. Die obere Tastenreihe entspricht den roten Knöpfen, die untere den grünen. Wird eine der Tasten betätigt, so ist die momentane Weichenstellung auch am Gleisbild zu erkennen. Gleichfalls wird beim Stellen einer Fahrstraße, das mit den 4 quadratischen Tasten an den Enden der Abstellgleise geschieht, auch das Stellpult aktualisiert. Wofür man die Tasten STOP und GO verwendet, braucht wohl nicht extra erläutert zu werden.

Um beim Start des Programms eine festgelegte Ausgangssituation zu schaffen, werden automatisch einige Vorarbeiten ausgeführt. Nachdem durch zweimaligen GO-Befehl das System sicher aktiv sein dürfte, werden zuerst alle Weichen in Default-Stellung gebracht und – soweit programmiert – die Funktionen bei den Loks aktiviert (im Beispielprogramm bleibt nur bei der Telex-lok 08 die Funktion aus). Im Kopf des Programms sind die fünf voreingestellten Loknummern in DATA-Zeilen enthalten, denen jeweils ein Flag für die Funktion folgt.



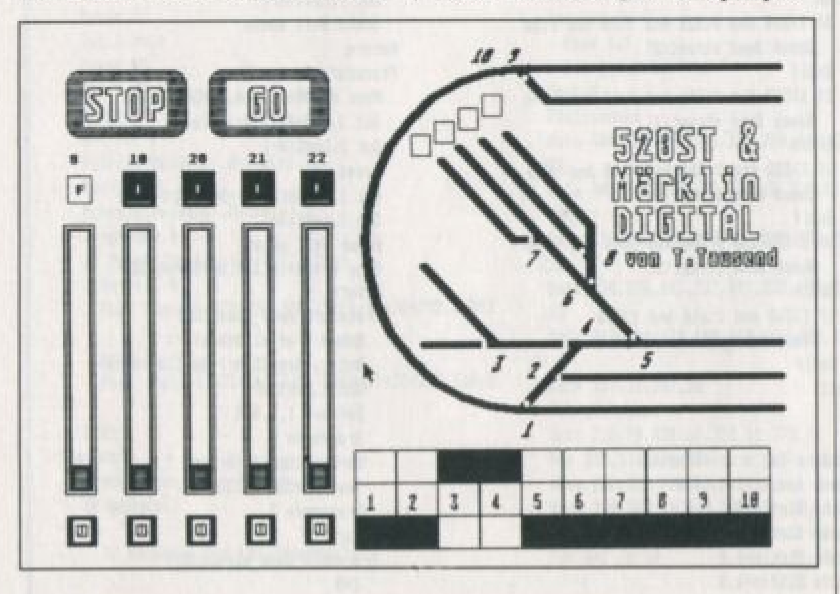
Für die Verbindung der Zentraleinheit des Märklin-Systems mit dem Computer sorgen Sie so.

Die Fahrwege enthalten jeweils Richtung und Weichennummern der zu schaltenden Fahrstraßen. Das Flag −1 zeigt das Ende einer Folge an. Unter Default stehen die Stellungen, die die Weichen 1-9 und das Signal (10) beim Programmstart einnehmen sollen.

Die verwendeten Schieberegler existieren in Wirklichkeit nur einmal; das Objekt wird erst dann an die entsprechende Stelle geschrieben, wenn man die Maustaste im Reglerbereich drückt.

Die Prozedur TEST, die ziemlich am Ende des Programms steht, ermöglicht das Anpassen an eigene Gleisbilder. Wenn man vor der Hauptschleife für die Bedienung ein GOSUB TEST einsetzt, so werden bei aufgebautem Bildschirm die jeweiligen Mauskoordinaten ausgegeben. Diese können dann zum Beispiel in das Datenfeld GPOS übernommen werden, wo jeweils X- und Y-Koordinaten für den Punkt im geraden und gebogenen Teil einer Weiche abgelegt werden.

Sollte dieser Bericht bei Computerfans und/oder Eisenbahnern auf entsprechende Resonanz stoßen, so werden wir in einer der nächsten Ausgaben des **ATARI *magazins*** ein Konstruktions-Set für ein Gleisbild-Stellwerk veröffentlichen.



Das Schaltbild für die komplette Steuerung per Computer

GFA-Basic-Programm

```
Loknummern:
Data 80,0,10,1,20,1,21,1,22,1
Fahrwege:
Data 34,5,33,6,34,7,32,-1
Data 34,5,33,6,33,7,32,-1
Data 34,5,34,6,34,8,32,-1
Data 34,5,34,6,33,8,32,-1
'
Dim R$(4),F$(4),Lok%(4),Bel%(4),Speed%(4),Bx(10),B
y(10),Rx(10),Ry(10)
'
Default: ! Stand der Weichen beim Systemstart
Data 33,33,34,34,33,33,33,33,33,33
Gosub Set_rs232 ! 2400 Baud einstellen...
Out 1,96 ! Freigabe senden
Pause 10
Out 1,96 ! sicher ist sicher...
Pause 10
'
Objdaten: ! für Schieberegler
Data -1,1,1,20,0,16,2,&1100,100,140,20,200
Data 0,-1,-1,20,32,0,3,&1121,1,100,10,20
Data -999
'
Gosub Make_screen
Restore Objdaten
Do
  Exit If A=-999
  Read A
  A$=A$+Mki$(A)
Loop
Restore Fahrwege
For N=0 To 3
  Do
    Read W
    Exit If W=-1
    F$(N)=F$(N)+Chr$(W)
  Loop
Next N
Aadr=Varptr(A$)
Restore Loknummern
For N=0 To 4
  Pos=N*50+20
  Dpoke Aadr+16,Pos
  R$(N)=A$
  Gosub Obj_draw(Aadr)
  Defline 1,1
  Box Pos,100,Pos+20,120
  Deftext 1,1,0,6
  Text Pos+7,113,"F"
  Read Lon
  Lok%(N)=Lon
  Read Fun
  If Fun=1
    Graphmode 3
    Gosub Send_f(N)
    Graphmode 1
  Endif
  Deftext 1,0,0,6
  Text Pos+2,90,Str$(Lon)
  Defline 1,3
  Box Pos,360,Pos+20,380
  Deftext 1,17,0,6
  Text Pos+6,373,"U"
Next N
Graphmode 3
Restore Default
For N=1 To 10
  Read Richtung
  If Richtung=34
    Gosub Send_rund(N)
  Else
    Gosub Send_gerade(N)
  Endif
  Pause 20
Next N
Pause 10
Out 1,32 ! Spulen abschalten
Deffill 1,2,8
Do ! Hauptschleife für Bedienung
  While Mousek<>1
    If Inkey$=" "
      If Stopflag=1
        Gosub Go
      Else
        Gosub Stop
      Endif
    Endif
  Wend
  Mouse X,Y,K
  If X>10 And X<240
    If Y>20 And Y<60
      If X>20 And X<120
        Gosub Stop
      Endif
      If X>140 And X<240
        Gosub Go
      Endif
    Endif
    If Y>77 And Y<92
      Gosub Lon_input(Int((X-10)/50))
    Endif
    If Stopflag=0
      If Y>140 And Y<360
        Sn=Int((X-10)/50)
        Gosub Send_speed(Sn)
      Endif
      If Y>100 And Y<120
        Fn=Int((X-10)/50)
        Gosub Send_f(Fn)
      Endif
      If Y>360
        Un=Int((X-10)/50)
        Gosub Send_u(Un)
      Endif
    Endif
  Else
    If X>310 And X<326 And Y>70 And Y<86
      Gosub Send_street(0)
    Endif
    If X>330 And X<346 And Y>60 And Y<76
      Gosub Send_street(1)
    Endif
    If X>350 And X<366 And Y>50 And Y<66
      Gosub Send_street(2)
    Endif
    If X>370 And X<386 And Y>40 And Y<56
      Gosub Send_street(3)
    Endif
    If X>260 And X<610 And Y>310
      Gosub Send_pult
    Endif
  Endif
Loop
End
Procedure Obj_draw(Adresse)
  Gosub Aesctrl(6,1,1,0)
  Dpoke Gintin,0
  Dpoke Gintin+2,9
  Dpoke Gintin+4,0
  Dpoke Gintin+6,0
  Dpoke Gintin+8,639
  Dpoke Gintin+10,400
  Lpoke Addrin,Adresse
  Gemsys 42
Return
Procedure Graf_slidebox(Nr)
  Radr=Varptr(R$(Nr))
  Gosub Aesctrl(3,1,1,0)
  Dpoke Gintin,0
  Dpoke Gintin+2,1
  Dpoke Gintin+4,1
  Lpoke Addrin,Radr
  Gemsys 76
  Rw=Dpeek(Gintout)
  Aktpos=Int((Rw+9)/5.5555)
  Dpoke Radr+42,Aktpos
  Speed%(Nr)=14-Rw/66.666667
  Gosub Obj_draw(Radr)
Return
Procedure Make_screen
  Cls
  Deffill 1,2,2
  Prbox 20,20,120,70
  Prbox 140,20,240,70
  Deftext 1,16,0,32
  Text 35,57,70,"STOP"
  Text 155,57,70," GO "
  Gosub Draw_tracks
Return
Procedure Stop
  If Stopflag=0
    Out 1,&H61
    Prbox 10,10,122,72
    Let Stopflag=1
  Endif
Return
Procedure Go
  If Stopflag=1
    Out 1,&H60
    Prbox 10,10,122,72
    Let Stopflag=0
  Endif
Return
Procedure Send_f(Nr)
  Bel%(Nr)=Not Bel%(Nr)
  Pbox Nr*50+16,96,Nr*50+44,124
  Out 1,(Speed%(Nr)+Bel%(Nr)*-16)
  Out 1,Lok%(Nr)
  Gosub Wait_mouse
Return
Procedure Send_u(Nr)
  Pbox Nr*50+16,360,Nr*50+44,380
  Out 1,(15+Bel%(Nr)*-16)
  Out 1,Lok%(Nr)
  Pause 10
  Out 1,(Speed%(Nr)+Bel%(Nr)*-16)
  Out 1,Lok%(Nr)
  Gosub Wait_mouse
  Pbox Nr*50+16,360,Nr*50+44,380
  Return
  Procedure Send_speed(Nr)
    Gosub Graf_slidebox(Nr)
    Out 1,(Speed%(Nr)+Bel%(Nr)*-16)
    Out 1,Lok%(Nr)
    Deftext 1,0,0,4
    Graphmode 1
    H$=Str$(Speed%(Nr))+" "
    Text Nr*50+23,135,H$
    Graphmode 3
  Return
  Procedure Send_street(Nr)
    I=0
```

```
  Do
    Richtung=Peek(Varptr(F$(Nr))+I)
    Exit If Richtung=32
    Inc I
    Wein=Peek(Varptr(F$(Nr))+I)
    Inc I
    If Richtung=34
      Gosub Send_rund(Wein)
    Else
      Gosub Send_gerade(Wein)
    Endif
    Pause 20
  Loop
Return
Procedure Send_pult
  Wein=Int((X-260)/35)+1
  If Y<360
    Richtung=34
    Gosub Send_rund(Wein)
  Else
    Richtung=33
    Gosub Send_gerade(Wein)
  Endif
Return
Procedure Send_rund(Wein)
  Out 1,34
  Pause 5
  Out 1,Wein
  Pause 15
  Out 1,32
  Graphmode 1
  Deffill 0
  Pcircle Gx(Wein),Gy(Wein),3
  Deffill 1
  Pcircle Rx(Wein),Ry(Wein),3
  Graphmode 3
  If Point((Wein-1)*35+265,315)=0
    Pbox (Wein-1)*35+260,310,(Wein-1)*35+295,310+2
5
    Deffill 0
    Pbox (Wein-1)*35+260,360,(Wein-1)*35+295,360+2
5
    Deffill 1
  Endif
Return
Procedure Send_gerade(Wein)
  Out 1,33
  Pause 5
  Out 1,Wein
  Pause 15
  Out 1,32
  Graphmode 1
  Deffill 1
  Pcircle Gx(Wein),Gy(Wein),3
  Deffill 0
  Pcircle Rx(Wein),Ry(Wein),3
  Graphmode 3
  If Point((Wein-1)*35+265,315)=1
    Deffill 0
    Pbox (Wein-1)*35+260,310,(Wein-1)*35+295,310+2
5
    Deffill 1
    Pbox (Wein-1)*35+260,360,(Wein-1)*35+295,360+2
5
  Endif
Return
Procedure Lon_input(Nummer)
  Graphmode 1
  Do
    If Mousek=1 And Lok%(Nummer)>0
      Dec Lok%(Nummer)
    Endif
    If Mousek=2 And Lok%(Nummer)<80
      Inc Lok%(Nummer)
    Endif
    Exit If Mousek=3
    Deftext 1,0,0,6
    Text Nummer*50+22,90,Str$(Lok%(Nummer))+" "
    Pause 4
  Loop
  Graphmode 3
Return
Procedure Wait_mouse
  While Mousek=1
  Wend
Return
Procedure Draw_tracks
  Deftext 1,16,0,32
  Text 400,90,"520ST &"
  Text 400,120,"Märklin"
  Text 400,150,"DIGITAL"
  Deftext 1,16,0,13
  Text 490,170,"von T.Tausend"
  Defline 1,5,2,2
  Circle 400,150,130,900,2700
  Draw 400,20 To 620,20
  Draw 400,20 To 425,45 To 620,45
  Draw 400,280 To 620,280
  Draw 400,280 To 450,230
  Draw 425,255 To 620,255
  Draw 320,230 To 620,230
  Draw 380,230 To 320,170
  Draw 500,230 To 350,80
  Draw 420,150 To 395,150 To 335,90
  Draw 460,190 To 460,140 To 390,70
  Draw 460,165 To 370,75
  Restore Positionen
  Deftext 1,4,0,13
  For N=1 To 10
    Read X,Y
    Text X,Y,Str$(N)
  Next N
  Restore Gbpos
  For N=1 To 10
    Read Gx(N),Gy(N),Rx(N),Ry(N)
  Next N
  Defline 1,1
  For N=1 To 4
    Read X,Y
    Box X,Y,X+16,Y+16
  Next N
  Positionen:
  Data 400,300,405,255,375,250,450,222,500,240,435
,200
  Data 405,170,470,170,390,15,360,15
  Gbpos:
  Data 412,280,405,275,431,249,432,255,370,230,375
,225
  Data 430,230,443,237,491,230,495,225,451,182,460
,182
  Data 414,144,410,150,460,157,453,159,409,20,406,
27
  Data 387,22,387,30
  Fahrstrassen:
  Data 310,70,330,60,350,50,370,40
  Box 260,310,610,385
  Draw 260,335 To 610,335
  Draw 260,360 To 610,360
  Deftext 1,2,0,13
  For N=1 To 10
    Draw 260+N*35,310 To 260+N*35,385
    Text 235+N*35,355,Str$(N)
  Next N
  Deffill 1,2,0
  Pbox 260,310,610,335
Return
Procedure Test
  Do
    Print At(10,10);Mousex;" ";Mousey;"   "
    Exit If Mousek=2
  Loop
  Error 7
Return
Procedure Set_rs232
  Void Xbios(15,4,0,-1,-1,-1,-1)
Return
Procedure Aesctrl(P1,P2,P3,P4)
  Dpoke Gcontrl+2,P1
  Dpoke Gcontrl+4,P2
  Dpoke Gcontrl+6,P3
  Dpoke Gcontrl+8,P4
Return
```



Going to San Francisco

Zu Besuch bei der amerikanischen Atari-Zeitschrift ANTIC

Nach einem langen Fußmarsch durch San Francisco stehe ich vor einem alten Backsteinbau. Kann es sein, daß der Welt größte Atari-Zeitschrift statt in einem modernen Wolkenkratzer in dieser Bruchbude residiert? Nun, das Bild, das sich mir nach Öffnen der Eingangstür bietet, steht ganz im Gegensatz zur tristen Fassade. Schreibmaschinengetippe, Druckerrattern, moderne Einrichtung und ein bombastisches ANTIC-Logo als Neonleuchte über dem Empfang. Alte ANTIC-Cover und ein großes "Atari at the Olympics"-Banner zieren die Wände. Ein Zeitschriftenständer zeigt ANTICs Produktpalette. Sie umfaßt außer

Unscheinbar nur von außen: Das ANTIC-Domizil in San Franzisco



dem gleichnamigen Magazin noch STart (für die 16-Bit-Ataris) und II-Computing, das Informationen für die Apple-User enthält.

Nun werde ich in Gary Yost's Büro gerufen. Gary Yost arbeitete früher als Toningenieur und hat mehrere bekannte Bands auf Tourneen begleitet. Auf seine Bitte hin wurde ihm in Ataris Frühphase eine Computeranlage geschenkt. Er begann, für Atari Software zu testen, wurde später an ANTIC vermittelt und begleitet dort seit 1983 den Posten eines Marketing Directors. Seine Aufgabe ist es, alle Programme, die ANTIC angeboten werden, genau zu prüfen, Verhandlungen mit den Programmierern zu führen und in der ganzen Welt die Rechte an interessanten Softwa-re-Neuheiten einzukaufen.

Beim Eintreten fällt mein Blick auf einen ST, dessen Monitor für diesen Rechner ungewöhnliche Dinge zeigt. Raumschiffe, Becher, Würfel und viele andere Gegenstände rotieren in bester 3-D-Manier auf dem Bildschirm. Keine Gittergrafik, sämtliche Flächen sind ausgefüllt. Man könnte fast meinen, einen AMIGA vor sich zu haben.

Auf meine Frage, ob hinter allem der neue Blitterchip steckt, winkt Gary ab. ANTICs neues CAD-Programm funktioniert auf Fractal-Basic und erlaubt so einen blitzschnellen Bildaufbau. Besitzer eines 8-Bit-Atari können diese Technik an den Lucasfilm-Spielen beobachten. Gary kam gerade aus England zurück, wo er die Amerika-Rechte für "Collossus-Chess", "Chop Suey", "Timebandit" und anderes mehr erwarb.

Meine kurze Audienz bei Gary Yost ist beendet. Nun empfängt mich Nat Friedland, ein ehemaliger Billboard Musikjournalist, der den Posten des Chefeditors innehat. Er erzählt mir ANTICs Entstehungsgeschichte.

Sie begann damit, daß Jim Caparell, ein NASA-Programmierer, in Marin County, Cal., ABACUS gründete, eine der ersten Atari-User-Groups. Als die Fangemeinde mehr und mehr wuchs, faßte er 1982 den Beschluß, ein Atari-Magazin herauszugeben. Als Namen wählte er die Bezeichnung eines der Grafik-Chips des Computers. Als Redaktionssitz diente sein kleines Appartement in San Francisco. Obwohl blutige Laien im Zeitschriftengeschäft, fanden Jim und seine Mitarbeiter aufgrund der in ganz Amerika herrschenden Computer-Euphorie rege Unterstützung. Zu den ersten Inserenten zählten kleine Firmen wie Synapse, Broderbund und Sierra On-Line. Der Riese Atari nahm noch wenig Notiz von ANTIC. Die erste Auflage betrug bescheidene 13.000 Exemplare.

1983 zog die Redaktion um. Sie residierte jetzt über einer zwielichtigen Bar im Havenviertel von San Francisco. Auch der neue Chefeditor stammte aus der Schiffsbranche, bestellte auf Kosten von ANTIC Schiffsmodelle, trank viel und wurde bald vor die Tür gesetzt. Ende 1983 umfaßte das Magazin 148 Seiten, und die Auflage stieg schnell.

Anfang 1984 folgte der nächste Umzug, nun in eine alte Kleiderfabrik, den heutigen Standort. Mitte desselben Jahres kaufte Jack Tramiel die fast bankrotte Firma Atari. Auch an ANTIC ging die Computerrezession nicht spurlos vorbei. Um das Magazin attraktiver zu machen, beschloß man einige Neuerungen. So konnten die Leser jetzt sämtliche Programme eines Heftes auf Diskette erhalten (heute bei uns selbstverständlich). Durch ANTIC ON-LINE wurde es für jeden Interessierten möglich, die

neuesten Nachrichten aus der Computerbranche, Tips und Tricks via Modem frei Haus auf den Bildschirm zu bekommen. Im Worldwide Users Network (WUN) wurde der Informationsaustausch zwischen User-Groups in aller Welt gefördert.

Nachdem sich Atari aus dem Software-Geschäft zurückgezogen hatten, übernahm ANTIC den Atari Program Exchange (APX) und baute ihn weiter aus. Heute vertreibt ANTIC die genannten drei Magazine, ca. 250 Programme, unterstützt 350 User-Groups und hat weltweit etwa 150.000 Leser.

Ich bedanke mich bei Nat und sehe mich noch ein wenig in der Redaktion um. Nahezu alle veröffentlichten Atari-Programme sind hier vertreten. Ein paar Neuheiten befinden sich im Test. "Worldchampionship Karate" von Epyx (bei uns "International Karate"), ein neues Textverarbeitungsprogramm, das laut Tester selbst "Paperclip" in den Schatten stellt, und "Cardiac Arrest", ein Lernprogramm für Mediziner, finden den Beifall der Redaktion.

Ein letzter Abstecher gilt ANTICs hauseigener Software-Abteilung. Am Anfang stand hier der Vertrieb von Public-Domain-Sofware (Programme, für die der Autor kein Copyright beansprucht und die beliebig kopiert werden dürfen). Später füllten Produkte den Katalog, die sonst niemand vertreiben wollte, weil sie nur für Spezialisten gedacht waren oder an der Grenze Profi-/Amateur-Software lagen. Manche dieser Programme kamen später zu Ruhm, wie z.B. "Dandy Dungeon", das die Vorlage für den Hit "Gaunlet" bildete. Auch viele heute bekannte Programmierer wie Stan Ockers, Chris Crawford oder Russ Wetmore taten hier ihre ersten Schritte.

Der Katalog bietet für jeden etwas. Er enthält eine Reihe von Strategiespielen, ein C-System, Musik- und Malprogramme sowie viele Utilities, die andere Software wie z.B. den "Printshop" ergänzen. Die ST-Sparte bringt dem europäischen User wenig Neues. Hier findet man "Timebandit", "Ultima II" und "CAD 3D", um ein paar Namen zu nennen.

Auf einige Programme möchte ich hier näher eingehen. "Earthviews" ist eine Weltkarte im Computerformat. Mit dem Cursor kann man die Welt durchstreifen, wobei der Computer immer den Namen der Gegend nennt, in der man sich befindet. Bei der Direkteingabe von Städten, Flüssen usw. zeigt der Rechner den jeweiligen Ort auf der Karte. Meine Heimatstadt Fürth ist leider im Speicher nicht enthalten, wohl aber die Nachbarstadt Nürnberg (hier natürlich: Nuremberg). Verschiedene Kartendarstellungen sowie ein rotierender Globus sind möglich. Wer glaubt, in Geographie fit zu sein, darf in einem kleinen Spiel seine Kenntnisse testen. Besitzer eines Atari-Cassettenrecorders können ihre Weltreise musikalisch unterlegen.

"Space-Base" führt in den Weltraum. Auf der scrollenden Sternenkarte läßt sich mit dem Cursor nahezu jeder größere Stern anwählen. Per Knopfdruck erhält man wichtige Daten und ein Hertzsprung-Russell-Diagramm. Dieses Programm wird Hobby-Astronomen erfreuen.

Spielerisch geht es beim "King Tut's Tomb Construction Set" zu. Hier durchforscht man als Schatzsucher dunkle Grüfte, immer auf der Flucht vor deren gefährlichen Bewohnern. Das Ganze stellt eine gelungene Umsetzung des Arcade-Spiels "Tutenkham" dar (schlägt die Parkerversion um Längen).

Weitere ANTIC-Programme werde ich demnächst im **ATARI** ***magazin*** ausführlich besprechen.

Nun möchte ich noch ein paar Public-Domain-Schmankerl von ANTIC vorstellen. "DOS 4.0" besitzt einen ähnlichen Aufbau wie "DOS 2.0", ist aber viel schneller, unterstützt single, enhanced and real double density und enthält viele Extras. Eine gute Dokumentation auf der Diskette erläutert sämtliche Funktionen. "DOS 4.0" wurde von Atari für viel Geld entwickelt, kam aber nie auf den Markt. Der Autor Michael Barall erhielt die Rechte zurück und beschloß, sein Werk allen Atari-Fans zur Verfügung zu stellen.

Freundlicher Empfang bei der größten Atari-Zeitschrift der USA



Die "ANTIC-Games-Disk #1" bietet eine Fülle von älteren Basic-Spielen, die zum Teil in ANTIC veröffentlicht wurden. Darunter befinden sich Stan Okkers "Chicken" und Klassiker wie "Hangman" und "Lunar-Lander".

Wer an diesen beiden Public-Domain-Programmen interessiert ist, kann sie von mir erhalten. Dazu überweisen Sie bitte 10 DM auf das Konto Nr. 1302918909 der Noris Bank Fürth, Bankleitzahl 76020400. Alle Leser, die den ANTIC-Katalog haben möchten, schreiben bitte an ANTIC Customer Service, 524 Second St., San Francisco, CA 94107, USA.

Der Katalog liegt auch jeder ANTIC-Ausgabe bei. Das Magazin ist in vielen Bahnhofsbuchhandlungen, bei Montanus oder diversen Software-Handlungen (z.B. Compy-Shop) erhältlich. Compy-Shop wird eventuell auch demnächst ANTIC-Software anbieten. Bei Fragen schreiben Sie bitte an die Redaktion des **ATARI** ***magazins***.

Frank Emmert



Mit GEM gegen die Pfunde

"Koch & Fitness" bringt Ordnung in die Rezeptssammlung und ist ein unbestechlicher Wächter über die Kalorien.

Mittlerweile gibt es für die ST-Computer ja bereits zahlreiche Programme aller Art. Nach aktuellen Informationen aus verschiedenen Softwarehäusern wird dieser positive Trend auch weiter anhalten.

Etwas aus dem Rahmen des Üblichen fällt das hier vorgestellte Programm "Koch & Fitness". Dabei handelt es sich nicht um ein Spiel, sondern, wie der Titel schon vermuten läßt, um eine Art elektronisches Kochbuch mit integrierter Kalorientabelle.

Voraussetzung ist ein ST-Computer mit einer Speicherkapazität von 1 MByte. Die Monitorausführung spielt keine Rolle, da das Programm sowohl mit einem Monochrom- als auch mit einem Farbmonitor (in der mittleren Auflösung) arbeitet. "Koch & Fitness" läuft vollständig unter GEM und bietet dem Anwender damit Benutzerfreundlichkeit und extrem kurze Einarbeitungszeit.

Nach dem Laden erscheint das Desktop-Bild auf dem Monitor. Dem Anwender stehen jetzt verschiedene Optionen zur Verfügung. Neben einigen Rezepten, die bereits auf der Diskette abgespeichert sind, kann man eigene eintragen, ändern, löschen, drucken und anschauen. Die einzelnen Rezepte werden alphabetisch geordnet angezeigt und bei Bedarf in Form einer Karteikarte grafisch dargestellt. Vermerkt sind dort alle Zutaten mit der entsprechenden Mengenangabe, die Kalorien- und Joule-Werte sowie die Zubereitungsanleitung. Sämtliche Rezepte können auch ausgedruckt werden.

Umfangreicher als die Rezeptsammlung ist die Kalorientabelle. Auch hier können weitere Einträge vorgenommen werden. Als Besonderheit hat man die Möglichkeit, eine persönliche Statistik zu erstellen. Gemeint ist damit die Erfassung der täglichen Gewichtszu- oder abnahme sowie der verbrauchten Kalorien. Wer gerade abnehmen will, hat damit ein schönes Kontrollinstrument zur Hand, da pro Tag nur ein Eintrag erlaubt wird und schönfärberische Manipulationen nicht möglich sind. Gerade mit einer solchen Kontrolle ist eine Abmagerungskur wahrscheinlich einfacher durchzuhalten als ohne.

Insgesamt betrachtet bietet "Koch & Fitness" eine Menge Möglichkeiten rund ums Essen. Leider ist der Preis ziemlich hoch. Wer sich davon nicht abschrecken läßt, kann demnächst auf die bekannte Zettelwirtschaft in der Küche verzichten.

System: Atari ST (min. 1 MByte)
Bezugsquelle:
K & K Software, München
Preis: ca. 169.– DM

Rolf Knorre

Statistik anlegen

Bitte geben Sie Ihren Namen, Ihr persönliches Passwort und das Startdatum ein.

Name :

Passwort:

Datum: 06/02/86

OK | Irrtum

Kalorienstatistik

Desk Rezepte Tabelle Statistik Optionen

KALORIEN-TABELLE ---- zeigen

	Hähnchenleber	100g	145 Kcal	607 Joule
E	Hähnchenkeule	100g	120 Kcal	502 Joule
E	Hähnchen	100g	145 Kcal	607 Joule
E	Hähnchenbrust	100g	110 Kcal	460 Joule
F	Hackepeter	100g	360 Kcal	1508 Joule
	Hackfleisch	100g	210 Kcal	879 Joule
K	Haferflocken	100g	400 Kcal	1676 Joule
V	Hagebutten	100g	105 Kcal	439 Joule
F	Hartwurst	100g	520 Kcal	2178 Joule
E	Harzerkäse 10%	100g	190 Kcal	796 Joule
E	Hase	100g	125 Kcal	523 Joule
F	Haselnüsse	100g	695 Kcal	2912 Joule
E	Hecht	100g	90 Kcal	377 Joule
	Hefewürfel	42g	40 Kcal	167 Joule
V	Heidelbeeren	100g	60 Kcal	251 Joule
E	Heilbutt geräu.	100g	240 Kcal	1005 Joule
E	Heilbutt	100g	110 Kcal	460 Joule

A B C D E F G H I J K L M N O P Q R S T U V W X Y Z

Das Programm hält eine komplette Kalorientabelle bereit.

Febr./März 1987 DM 5.50 ÖS 48 SF 5.50

CK Computer Kontakt

Die User Zeitung für Atari, Sinclair und TI 99/4A

Nr. 2/3 4. Jahrgang

Sinclair

- ★ QL-Peintre
- ★ ZX-81-Biorhythmus
- ★ Spectrum-Interscript
- ★ Spielelisting: The Race

Atari

- ★ Spielelisting: Bergmann
- ★ Alarm-Tuner
- ★ Topprogramm: Textverarbeitung
- ★ Elisa

TI 99/4A

- ★ CPS im Dauertest
- Spielelistings: Mau, Wator



Für alle kleinen Ataris:

Noch mehr Spielreviews, Listings sowie Tips + Tricks im neuen Computer-Kontakt – an Ihrem Kiosk

Der Star wird bunt

Nicht nur die Farbe macht aus MonoStar den ColorStar.



Für ein Grafikprogramm selbstverständlich: die Lupenoption.

Im letzten **ATARI** ***magazin*** haben wir das Grafikprogramm "Monostar" vorgestellt. Jetzt liegt mit "Colorstar" die Farbversion aus dem gleichen Haus vor, die wir natürlich auch getestet haben. Hier das Ergebnis.

16 Bit

Beide Programme stammen vom selben Autor. Daher läuft die Bedienung von "Colorstar" in gleicher Weise ab wie die von "Monostar", ist also voll mausgesteuert. Identisch sind auch viele Grundfunktionen, auf die ich deshalb hier nicht mehr eingehen möchte. Das gilt für alle normalen Zeichenoptionen wie Freihand, Rechteck, Kreis usw., aber auch für Besonderheiten wie Spiegeln, Glätten und sogar Biegen. Auch die LOAD- und SAVE-Optionen für Bilder und Objekte sowie die Möglichkeit, drei Bilder gleichzeitig im Speicher zu halten, sind vorhanden.

Allerdings läßt sich hier keine komplette DIN-A4-Seite bearbeiten. Dafür sind einige neue Optionen hinzugekommen, die es in sich haben. Außerdem bietet "Colorstar" eben Farbe an; in der niedrigauflösenden Darstellung, die der Programmierer gewählt hat, steht immerhin eine Palette von 16 Farben zur Verfügung. Dazu später mehr. Kommen wir jetzt erst einmal zu den Neuheiten bei der Bildbearbeitung.

Die Funktion FORM ermöglicht das stufenlose Vergrößern und Verkleinern von Bildteilen, die man – wie bei fast jeder anderen Funktion auch – vorher mit einem Gummiband einrahmen kann. Auch der gesamte Bildschirminhalt kann beliebig verkleinert werden.

PFLASTER ist eine Vervielfältigungsfunktion, mit der sich interessante Effekte, auch in der Hintergrundgestaltung, erzielen lassen. Ein eingerahmter Bildteil wird im Extrabild so oft kopiert, bis der ganze Monitor gefüllt ist.

Der manuellen Bearbeitung dient die Option SCHMIERE. Mit ihr können harte Bildübergänge durch Verschmieren weicher gestaltet werden.

Die Funktion RASTER ist eine reine Effektoption. Ein eingerahmter Bildteil wird aufgerastert. Dadurch wirkt der Ausschnitt leicht unscharf und weist horizontale Verzerrungen auf.

Wesentlich interessanter ist die Option KIPPE, mit der sich dreidimensionale Effekte erzeugen lassen. Nachdem man z.B. einen Schriftzug eingerahmt hat, kann man durch Bewegungen der Maus Richtung und Stärke der Kippung festlegen. Danach ist es noch möglich, den Bildteil zu justieren.

Ähnlich wird auch bei der Option TROMMEL verfahren, die die vielleicht spektakulärsten Verfremdungen ermöglicht. Mit der Maus lassen sich Radius und Neigungswinkel der Trommel bestimmen. Nach dem Verformen kann man die Trommel noch einmal drehen, um festzulegen, welcher Teil nach vorne und welcher nach hinten kommt. Erst dann wird der Vorgang abgeschlossen. Der richtige Umgang mit dieser Option erfordert einige Übung. Das macht aber so viel Spaß, daß man das Ausprobieren gar nicht mehr lassen kann.

Die letzte Spezialoption nennt sich DREHE und ermöglicht die Drehung eines Bildausschnitts um 90 Grad.

Es ist immer wieder erstaunlich, wie schnell all diese Optionen ausgeführt werden. Auch die



16 Farben stehen zur Verfügung. Natürlich kann die Palette geändert werden.

NEU: Eine Textverarbeitung – So gut, daß wir mit unserem Namen dafür stehen.

BECKER textST




```
Desk   Datei   Bearbeiten   Rechnen   Einstellungen   Schrift
BRIEF                       8 LPI  ZEILE  20  SPALTE  1

|POS|STÜCK|  BEZEIC...            | EZ.PR  | GES.PR |
| 1 |  2  | BECKERtext ST         | 199.00 | 398.00 |
| 2 |  1  | Karton Papier ei...   |  49.00 |  49.00 |
| 3 |  3  | Farbbänder EPSON      |  12.80 |  38.40 |
| 4 |  1  | DATAWELT              |   6.00 |   6.00 |
|   |  7  |                       | Summe  | 491.40 |

Rechnen:
  Spalte summieren
  Spalte subtrah.
  Zeile summieren
  Zeile subtrah.
  Register laden
  Register -> Text
  Register +
  Register -
  Register *
  Register /
  Register %
  Dezimalstellen
```

Der enorme Erfolg von TEXTOMAT ST spornte die Autoren dieser leistungsstarken Textverarbeitung an. Sie wollten nun ein Programm entwickeln, daß noch besser ist, noch mehr leistet als TEXTOMAT ST und Unmögliches möglich macht. Das Ergebnis gefiel uns so gut, daß wir ihm unseren Namen geben: BECKERtext ST. Natürlich finden Sie alle bewährten Features von TEXTOMAT ST wieder.

NEU: Schnelle Direktformatierung

Direktformatierung am Bildschirm und trotzdem schnelles Scrolling – unmöglich? – Doch möglich! Mit BECKERtext ST.

NEU: Mehr Zeichen am Bildschirm

80 Zeichen und noch breitere Texte voll sichtbar am Bildschirm bearbeiten – unmöglich? – Doch möglich! Sogar bis zu 96 Zeichen pro Zeile in bis zu 26 Zeilen mit BECKERtext ST.

NEU: ONLINE-Lexikon

Eine schnelle Rechtschreibhilfe, die wahlweise ONLINE während des Schreibens oder nachträglich den kompletten Text überprüft, die lernfähig ist und mit unterschiedlichen, individuell gestaltbaren Lexika arbeiten kann – unmöglich? – Doch möglich! Mit BECKERtext ST.

NEU: Komfortables Rechnen im Text

Komfortables Rechnen im Text nicht nur spalten- sondern auch zeilenweise mit bis zu 6 Nachkommastellen und 12stelliger Genauigkeit – unmöglich? – Doch möglich! Mit BECKERtext ST.

NEU: Formulare mit Komfort

Komplette Formulare unterschiedlichster Art als Eingabemasken erstellen, die leicht bearbeitet und nicht mehr zufällig überschrieben werden können – unmöglich? – Doch möglich! Mit BECKERtext ST.

Dazu natürlich viel Produktpflege mit Liebe zum Detail von 128 zusätzlichen Sonderzeichen über Dezimaltabulatoren bis zu einer großen Anzahl ausgeklügelter Druckeranpassungen.

BECKERtext ST kostet einschließlich ausführlichem Handbuch DM **199,–**

Einen ausführlichen Prospekt schicken wir Ihnen gerne kostenlos.
Lieferbar ab ca. Anfang Februar.

DATA BECKER

Merowingerstr. 30 · 4000 Düsseldorf · Tel. (0211) 310010

BESTELL-COUPON

Einsenden an: DATA BECKER · Merowingerstr. 30 · 4000 Düsseldorf 1

Bitte senden Sie mir:

BECKER text ST DM 199,–

zzgl. DM 5,– Versandkosten

☐ per Nachnahme ☐ Verrechnungsscheck liegt bei

Name

Straße

Ort



Mit der Verkleinerungsoption lassen Sie Bäume rückwärts wachsen.

daraus resultierenden Effekte kann man nur als toll bezeichnen.

Kommen wir jetzt zu zwei weiteren Änderungen gegenüber "Monostar". Zunächst sind hier die einzustellenden Verknüpfungsarten zu nennen. "Monostar" verfügt nur über FEST und KLAR. "Colorstar" bietet dagegen 13 verschiedene Verknüpfungsmöglichkeiten für Vorder- und Hintergrund an, mit denen man sich lange beschäftigen muß, um herauszufinden, was alles machbar ist.

Die zweite Neuheit betrifft die Farbgestaltung. Nach Aufruf der entsprechenden Menüoption erscheint auf dem Bildschirm ein Balken mit den 16 eingestellten Farben, aus denen man dann die gewünschte aussuchen kann. Wer mit der vorhandenen Palette nicht zufrieden ist, kann ein weiteres Farbmenü aufrufen. Alle Änderungen, die man hier vornimmt, wirken sich direkt auf das zuvor verlassene Bild aus.

Es ist möglich, von diesem Menü aus zwischen den drei Bildspeichern umzuschalten und sich diese anzeigen zu lassen. Im unteren Menüteil steht der Farbbalken, den man schon vorher gesehen hat. Jede Farbe läßt sich anklicken und durch Verschieben der drei Regler für Rot, Grün und Blau verändern bzw. mischen. Wem dieser Handbetrieb zu umständlich ist, der kann insgesamt 50 vordefinierte Paletten abrufen und sich das Ergebnis direkt im Bild ansehen.

"Colorstar" bietet außerdem noch die Funktion ZUFALL an. Damit kommt eine zufällig berechnete Farbpalette zur Darstellung. Die Funktion ÜBERGANG ermöglicht das Errechnen fließender Farbverläufe zwischen zwei einzustellenden Farben. Mit dieser Vielfalt der Farbgestaltung sollte eigentlich jeder Anwender zufrieden sein.

Der gute Eindruck, den "Monostar" hinterlassen hat, wird durch "Colorstar" voll bestätigt. Ich rechne beide Programme zu den besten Grafikverarbeitungssystemen, die zur Zeit für den ST verfügbar sind.

System: Atari ST, Farbmonitor
Hersteller/Bezugsquelle:
Brain Wave, Wuppertal

Rolf Knorre



Englisches Wetter für Atari

Amstrad sorgt für Turbulenzen im Höhenflug des ST

Hier bei uns war 1986 das Atari-Jahr, aber für Großbritannien traf dies nicht ganz zu. Die Herausforderung von Amstrad durch den neuen PC mit Werbekampagnen und Medien-Beeinflussung mag dazu beigetragen haben, zweifellos auch Personaländerungen auf höchster Ebene. Das Ergebnis war ein beträchtliches Durcheinander. Zeitweise war es so schlimm, daß sogar Atari selbst hinsichtlich des Geschehens verunsichert zu sein schien. Es hatte den Anschein, als ob praktisch jeden Monat neue Modelle mit einer verwirrenden Reihe von Serien-Nummern angekündigt würden, von denen die meisten nie verwirklicht wurden. Im Januar war es der 1040 ST, während der noch relativ junge 260 ST herausgenommen werden sollte. Der März brachte zwei neue Modelle, den 520 ST und den 520 STFM. Im April sollte der 520 ST durch den 1040 STF abgelöst werden. Die Möglichkeit eines 32-Bit-ST als Konkurrenz zum Amiga wurde im Juni angedeutet. Im September folgte eine Ankündigung an den Handel mit den Modellen 2M, 4M, 2080 STS und 4160 STS und im November wurde vertraulich vorausgesagt, daß der 520 STEM (Sie erinnern sich an den März?) im Januar erhältlich sein werde.

Unfair ist allerdings, daß man Amstrad zum Helden erklärte, während Atari als Amstrads einziger ernsthafter Konkurrent nur Tadel erntete. Schließlich hatte Amstrad die Existenz des PC 1512 derart massiv geleugnet, bis dann das neue Produkt doch erschien. Auch die Nachfrage nach den Hard-Disc-Modellen wurde derart falsch eingeschätzt, daß es zu einem sechsmonatigen Lieferverzug kam. Selbst auf der Personal-Computer-Weltshow im Olympia in London, wo Atari den bei weitem größten Stand mit 1000 m² innehatte, den die Firma mit 40 Lieferanten aus angrenzenden Bereichen teilte, wurde Atari in den Schatten gedrängt. Es gab eine derart massive Werbung für den ersten Auftritt des PC 1512 von Amstrad, der genau einen Tag vor Eröffnung der Show auf den Markt gebracht wurde.

Vor der Show meinte Ataris derzeitiger Marketing-Manager, Bob Harding, man wolle beweisen, daß der ST ein vielseitiges vertrauenswürdiges Produkt sei. Atari werde in der Hauptsache nicht über die zukünftige Technologie sprechen, sondern über die Technologie, die sie bereits hätten. Betrachtet man die Produktionsverzögerungen beim PC 1512, ist dies eine tiefsinnige Aussage!

Ein riesiges Schild lud die Besucher ein, "die Atari-Welt zu betreten", und viele der Besucher folgten der Aufforderung. Man befand sich in einem "Dorf", das gänzlich allen verschiedenen Produkten für die Atari-Computer gewidmet war, angefangen mit Spielen und Zubehör bis zu Handelsverpackungen und Büchern. Es waren einige der bedeutendsten Software-Hersteller vertreten. "Wild Bill" Stealey, der Direktor von Microprose war z.B. persönlich dabei. Er führte voller Begeisterung ein schwieriges Spiel für Erwachsene vor. Ein führender Journalist beschrieb den Atari ST als "den Computer, der allen die Show stahl". Trotzdem bleibt es eine Tatsache, daß auf der Show und während des ganzen Jahres Amstrad die Schlagzeilen galten. Wir werden sehen, ob 1987 Atari die Erfolge von vor zwei Jahren wiederholen kann – ein guter Start wäre der langerwartete 520 STFM.

Peter Cornforth

Animation der Zeichen

Teil 2 unserer Serie zur Spieleprogrammierung wird nun praktisch. Hinter der fantastischen Grafik vieler Spiele verbergen sich Zeichensatzmanipulationen. Ein Basic-Listing zeigt das Prinzip.

In der letzten Folge hatten wir uns damit beschäftigt, die optimale Programmiersprache für ein selbsterstelltes Spiel zu finden. Diesmal geht es um die Grafik, genauer gesagt, wir wollen uns ansehen, wo auf diesem Gebiet die Stärken und Schwächen des Atari liegen und welche Möglichkeiten er zur Realisierung guter Grafiken bietet.

Spielergrafik

Die Hardware des Atari enthält zwei unabhängige Schaltkreise zur Erzeugung der Bildschirmgrafik, die Spieler- oder PM-Grafik und die Hintergrund- oder Playfield-Grafik. Erstere bietet sich an, um schnell bewegte Figuren auf den Bildschirm zu zaubern. Sie wird bei anderen Computern auch oft als Sprite-Grafik bezeichnet und kann bis zu acht unabhängige Figuren in einer Bildschirmzeile darstellen. Allerdings sind vier dieser Objekte nur zwei Bildpunkte breit. Sie werden Missiles (Geschosse) genannt und sind jeweils einem der vier anderen (acht Punkte breiten) Objekte zugeordnet, die als Players bezeichnet werden.

Diese etwas merkwürdige Anordnung ist ein Ergebnis der frühen Spiele wie "Space-Invaders" oder "Gunfight", in denen rigoros geballert wurde und daher jede Spielfigur über ein Geschoß verfügen sollte.

Heutzutage hat man diese recht gewalttätigen Spiele hinter sich gelassen (wenigstens teilweise). Dies haben die Entwickler der Atari-Grafik-Chips anscheinend schon berücksichtigt. Die vier Missiles lassen sich nämlich auch zu einem fünften Player zusammenfassen.

Diese fünf Objekte unterliegen jedoch gewissen Einschränkungen hinsichtlich Größe und Farbe. Während ein Player in vertikaler Richtung die ganze Größe des Bildschirms einnehmen kann, beträgt die horizontale Auflösung nur acht Bildpunkte (Pixels). Pro Player ist auch nur eine einzige Farbe möglich. Zur Darstellung mehrfarbiger Spielfiguren muß man mehrere Players hintereinander anordnen. Soviel als Überblick; wir werden uns in einer der nächsten Folgen noch wesentlich intensiver mit dieser Thematik beschäftigen.

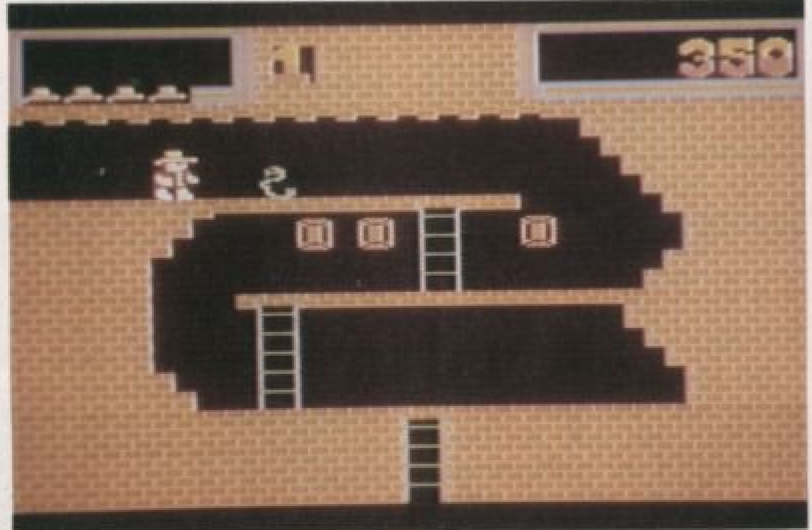
Playfield-Grafik

Die "normale" Bilddarstellung faßt man beim Atari unter dem Oberbegriff Hintergrund- oder Spielfeldgrafik zusammen. Gerade hier kann dieser Rechner mit seinen Grafik-Chips Antic und GTIA einen starken Trumpf ausspielen. Kein anderer Computer beherrscht so vielfältige Möglichkeiten der Darstellung. Antic kann nicht weniger als 14 verschiedene Grafikmodi erzeugen; zudem ist GTIA in der Lage, einige davon noch auf verschiedene Arten zu interpretieren. Diese Modi lassen sich grundsätzlich in zwei große Gruppen aufteilen, Bitmap- und Zeichensatzgrafik.

Bitmap-Grafik

Unter diesem Sammelbegriff versteht man Grafikmodi, bei denen der Zustand eines Punktes auf dem Bildschirm durch ein oder mehrere Bits direkt bestimmt wird. Hier herrscht eine große Vielfalt, beginnend bei dem sehr "blockigen" Modus GRAPHICS 3 bis hin zur hochauflösenden Grafikstufe 8.

Vielleicht werden Sie sich jetzt fragen, wozu man eine niedrige




Montezumas Rache ist ein gutes Beispiel für die Möglichkeiten der Zeichensatzgrafik.

Auflösung braucht, wenn man auch mit hoher arbeiten kann. Dafür gibt es zwei wichtige Gründe. Erstens sind in der höchstauflösenden Stufe nur zwei Farben verfügbar (genaugenommen nur 1 1/2), während niedrige Auflösungen wie z.B. in GRAPHICS 7 vier verschiedene Farben bieten. Zweitens kann im letzteren Fall der Speicherplatzbedarf wesentlich geringer zu Buche schlagen. Während GRAPHICS 8 fast acht KByte benötigt, kommt man in GRAPHICS 3 mit wenigen hundert Bytes aus. Bei einem gesamten Speichervolumen von 64 KByte ist das schon bedeutsam.

8 Bit

Natürlich bedingt die Speichereinsparung eine schlechtere Grafik. Einige Spiele wenden diese Strategie aber sehr geschickt an. Langjährige Atari-Besitzer werden sich sicher noch an "Galahad and the holy Grail" erinnern, im großen und ganzen das einzige Spiel, das GRAPHICS 3 recht effektiv einsetzte. Oder denken Sie an "Rescue on Fractalus" von Lucasfilm, das seine flugsimulatorähnlichen Sequenzen in atemberaubendem Tempo erzeugt. Dies ist nur möglich, da der speicherplatzsparende Modus 7 für den Blick aus dem Cockpit eingesetzt wird. Würde man stattdessen den doppelt so hoch auflösenden Modus 15 verwenden, so würde sich die Aufbaugeschwindigkeit der Bilder halbieren (da mehr Bytes bewegt werden müssen). Das hätte zweifellos einen ruckigen Ablauf zur Folge.

Bitmap-Grafik ist optimal für Spiele, in denen viele bewegte Objekte auftauchen oder die mit Liniengrafik arbeiten (z.B. "Mercenary"). Vielleicht erinnern Sie sich in diesem Zusammenhang auch an das Action!-Center der letzten Ausgabe. Grundsätzlich gilt, daß Bitmap-Grafik viel Speicherplatz in Anspruch nimmt und auch die Bewegung von Objekten sehr viel Rechenzeit benötigt.

Zeichensatzgrafik

Ganz im Gegensatz dazu steht die Grafik auf der Basis eines Zeichensatzes, die man auch oft Character-Grafik nennt. Hier lassen sich mit sehr geringem Speicheraufwand riesige Spielfelder unterbringen, von denen Teile mit Hilfe des fantastischen Atari-Scrollings am Bildschirm dargestellt werden können.

Der Unterschied zur Bitmap-Grafik liegt in einer Art Umweg. Jedes Byte des Bildschirmspeichers beschreibt jetzt nicht mehr nur einige Punkte auf dem Schirm, sondern wählt ein Zeichen eines Fonts (engl. für Zeichensatz) aus. Da ein Zeichen beim Atari zwischen 4×8 und 8×16 Bildpunkte beschreibt, behält der Videospeicher eines Zeichenmodus eine überschaubare Größe. Alles, was man zum Einsatz dieser Grafik braucht, ist ein Zeichensatzeditor, mit dem sich die gewohnten Buchstaben und Zahlen in Ufos, Männchen usw. verwandeln lassen.

Im "normalen" Modus arbeitet der Atari immer mit einer Zeichensatzgrafik (GRAPHICS 0), die zur Darstellung von Texten und Listings benutzt wird. Diese Stufe eignet sich für Spiele weniger, da die Farbauswahl wie in GRAPHICS 8 sehr eingeschränkt ist. Die Modi GRAPHICS 1 und 2 sind da schon wirkungsvoller, da sie den Einsatz von fünf Farben zulassen. Allerdings kann hier ein Zeichen nur zwei verschiedene dieser fünf Farben besitzen, was diese Modi für Spiele fast unbrauchbar macht. Sie lassen sich jedoch gut zur Anzeige des Punktestandes oder ähnlichem verwenden. Man muß sich beim Atari schließlich immer bewußt sein, daß es möglich ist, alle Grafikmodi beliebig zu mischen, gleichgültig, ob es sich um Bitmap- oder Zeichensatzgrafik handelt.

Wesentlich interessanter sind die Modi 12 und 13, die vierfarbige Zeichen zulassen. Insgesamt sind sogar fünf verschiedene Farben möglich, wobei die fünfte dem Zustand invers entspricht. Das bedeutet, daß eine Zeichensatzgrafik sogar farbenprächtiger ist als eine entsprechende Bitmap-Grafik und dabei noch Speicherplatz einspart. Ein Bild in GRAPHICS 12 belegt nur ca. 1 KByte Videospeicher und 1 KByte für den Zeichensatz. Die gleiche Auflösung als Bitmap-Grafik (Modus 15) nimmt dagegen ganze 8 KByte in Anspruch.

Wir wollen aber auch den Nachteil nicht vergessen: Da nicht unbegrenzt viele Zeichen zur Verfügung stehen, müssen sich gewisse Teile des Bildes gleichen. Dies ist jedoch in der Regel keine sehr bedeutsame Einschränkung, da die Erfahrung zeigt, daß die meisten Grafiken aus sich wiederholenden Mustern aufgebaut sind. Zur Verwendung der genannten Stufen benötigt man einen Zeichensatzeditor, der für mehrfarbige Grafiken geeignet ist.

Man kann guten Gewissens behaupten, daß die Mehrzahl der für den Atari erhältlichen Spiele aufgrund der geschilderten Vorteile diese Grafikstufen verwenden. Zu nennen sind hier Programme wie "Zeppelin", "Montezuma's Revenge", "Pharao's Curse", "Schreckenstein", "Bruce Lee" sowie Klassiker wie "Junge Hunt", "Frogger" und "Moon Patrol".

„Als die Bilder laufen lernten!"

Zum Abschluß dieser Folge wollen wir noch einen Blick auf eine einfache Art der Animation werfen, die sich bei Zeichensatz-

grafik förmlich anbietet. Der Grundgedanke ist folgender: Wenn man das Bitmuster eines Zeichens aus dem Zeichensatz abwandelt, ändern entsprechend alle Zeichen am Bildschirm sprunghaft ihr Aussehen. Verwendet man daher mehrere Zeichensätze, die sich periodisch abwechseln, so läßt sich permanente Bewegung mit minimalem Aufwand erzielen.

Wer ein Beispiel sehen möchte, kann Listing 1 in Basic ausprobieren. Es zeigt einen sprudelnden See mit Zufluß. (Die Ähnlichkeit mit "Cavelord" ist übrigens nicht ganz zufällig). Gearbeitet wird mit insgesamt vier Zeichensätzen, die durch die FOR-NEXT-Schleife am Ende des Programms mit Hilfe der Speicherzelle CHBAS (dez. 756) durchgeschaltet werden. Die Bitmuster der Zeichen sind in den DATA-Zeilen enthalten.

Sie werden sich hier wahrscheinlich wundern, denn in der letzten Folge habe ich die Programmiersprache Basic als für Animation untauglich eingestuft. Das ist im Prinzip auch richtig, aber die Animation in unserem Beispiel erfordert so wenig Rechenzeit, daß sich sogar Basic eignet. Es ist also keine Frage, wieviel Rechenzeit erst übrigbleibt, wenn man das Programm nicht in Basic, sondern in Assembler oder in Action! codiert.

In der nächsten Folge werden wir uns weitere Arten der Animation ansehen und einen Blick auf eine für die Spieleprogrammierung ganz wichtige Sache werfen, nämlich Interrupts. In diesem Zusammenhang darf ich Sie auch auf das Action!-Center in dieser Ausgabe verweisen. Viel Spaß bis zum nächsten Heft.

Peter Finzel

Demo in Basic

```
1000 REM ***************************
1010 REM Demo zur Zeichenanimation
1020 REM
1030 REM P. Finzel              1987
1040 REM ***************************
1050 DIM ZSATZ(4):CHBAS=756
1060 FOR I=0 TO 3
1070 ZSATZ(I)=(I+32)*1024
1080 NEXT I
1090 REM * Starre Zeichen
1100 FOR ZEICHEN=0 TO 1
1110 FOR I=0 TO 7:READ D
1120 FOR J=0 TO 3
1130 POKE ZSATZ(J)+ZEICHEN*8+I,D
1140 NEXT J:NEXT I
1150 NEXT ZEICHEN
1160 REM * Bewegliche Zeichen
1170 FOR ZEICHEN=2 TO 3
1180 FOR I=0 TO 3
1190 FOR J=0 TO 7:READ D
1200 POKE ZSATZ(I)+ZEICHEN*8+J,D
1210 NEXT J:NEXT I
1220 NEXT ZEICHEN
1230 REM * Bild zeichnen
1240 GRAPHICS 13+16:COLOR 33
1250 POKE CHBAS,ZSATZ(0)/256
1260 SETCOLOR 0,0,14:SETCOLOR 1,15,4
1270 PLOT 0,6:DRAWTO 0,11
1280 DRAWTO 39,11:DRAWTO 39,6
1290 COLOR 34
1300 FOR I=6 TO 10
1310 PLOT 1,I:DRAWTO 38,I
1320 NEXT I
1330 COLOR 35:PLOT 10,0:DRAWTO 10,5
1340 REM * Animationsschleife
1350 FOR I=0 TO 3
1360 POKE CHBAS,ZSATZ(I)/256
1370 FOR J=0 TO 25:NEXT J
1380 NEXT I
1390 GOTO 1340
1400 REM * ZEICHENSATZ
1410 DATA 0,0,0,0,0,0,0,0
1420 DATA 85,106,106,106,85,169,169,169
1430 DATA 255,223,253,127,255,223,253,127
1440 DATA 223,253,127,255,223,253,127,255
1450 DATA 253,127,255,223,253,127,255,223
1460 DATA 127,255,223,253,127,255,223,253
1470 DATA 63,63,220,252,253,63,255,247
1480 DATA 247,63,63,220,252,253,63,255
1490 DATA 255,247,63,63,220,252,253,63
1500 DATA 63,255,247,63,63,220,252,253
1510 DATA 85,106,106,106,85,169,169,169
```

1 Ohrenschmaus: Soundmachine

Vierstimmig, 10 Hüllkurven, Schlagzeug, bis zu 5000 Noten, auch von eigenen Programmen nutzbar, Eingabe über Tastatur oder Joystick. Mit Demos auf 2 Diskettenseiten, ausführliches Handbuch.
ATARI 400 - 130 XE, ab 48 K

Best.-Nr. AT 1 **29.80 DM**

2 Lehrreich: Das Assemblerbuch

Klare Einblicke in Zahlensysteme, in Aufbau und Befehlssatz des 6502, in Programmierung der Custom-Chips, Player-Missile-Grafik und Interrupt-Techniken sowie in einigen anderen Kauderwelsch. Dennoch für Einsteiger geeignet. Listings für ATMAS II Assembler. 196 Seiten DIN A5, direkt bei CK.

Best.-Nr. AT 10 **29.80 DM**

3 Vielfalt: Atari Power Superbuch

Bauanleitungen, Listings, Tips & Tricks ... 75 Seiten DIN-A4, nicht im Buchhandel erhältlich!

Best.-Nr. AT 3 **29.– DM**

4 Nachschlag: Die Hexenküche

Aufschlußreich für Ein/Aussteiger und Profis gleichermaßen: Tips & Tricks, Kniffe, Drehs etc. Maschinensprache-Programme als Listings. Turned Ihren Atari ganz schön an (und Sie auch)!

Best.-Nr. AT 4 **29.80 DM**

5 Zuschlag: Disk zu Hexenküche

Damit kann man viel Zeit sparen.

Best.-Nr. AT 5 **19.80 DM**

6 Fix: ATMAS-II Macro Assembler

8 K Quelltext in 4 Sekunden assembliert! Erzeugung von Bildschirmcode, Full-Screen Editor, scrollt in beide Richtungen, integrierter Monitor. 50 seitiges Handbuch und Disk im Ringordner.
ATARI 400 - 130 XE

Best.-Nr. AT 6 **Diskette 49.– DM**

7 Geistsparend: ATMAS Toolbox

Rechenroutinen, I/O Makros, Customizer, Fast circle, Scrolling und noch einiges mehr. Auf Diskette mit Anleitung daselbst
ATARI 400 - 130 XE, ab 48 K

Best.-Nr. AT 7 **19.80 DM**

8 Praktisch: Monitor XL

Verknüpft Basic-Programme mit Mcode-Routinen: eingeben, korrigieren, listen, Single-Step, Disk laden/speichern, Directory-Anzeige, deutsche Fehlermeldungen auch für Basic und DOS. Der Basic-Speicherplatz bleibt unberührt, Anleitung und Disk.
ATARI 600 XL (64 K) / 800 XL / 130 XE

Best.-Nr. AT 8 **19.80 DM**

9 Brandneu: Design Master

Bedienung über Fenster-Technik, Auflösung 320 • 192, Fadenkreuz, Maßstabsgitter ein/ausblenbar, 2 Screens gleichzeitig, über 122 000 Punkte im Direktzugriff, über 100 verschiedene Schriften, Hardcopy für fast alle Matrix-Drucker (ab 8 Nadeln), Ausdruck in verschiedenen Größen möglich, ausführliche deutsche Anleitung.
ATARI 600 XL (64 K) / 800 XL / 130 XE

Best.-Nr. AT 9 **Diskette 19.80 DM**

ATARI *magazin*

Spezialitäten-Bestellschein

Anzahl	Best.-Nr.	Artikel	
	AT 1	Soundmachine	
	AT 10	Das Assemblerbuch	
	AT 3	Atari Power Superbuch	
	AT 4	Die Hexenküche	
	AT 5	Disk zur Hexenküche	
	AT 6	Macro Assembler	
	AT 7	ATMAS Toolbox	
	AT 8	Monitor XL	
	AT 9	Design Master	
	AT 11	Diskmaster	

Ich wünsche folgende Bezahlung:
☐ Nachnahme (zuz. 5.70 DM Versandkosten)
☐ Vorauskasse (keine Versandkosten)
Bei Vorauskasse bitte Scheck beilegen oder auf Postscheckkonto Karlsruhe 434 23-756 überweisen.

Name des Bestellers ____________ PLZ/Ort ____________

Anschrift ____________ Datum/Unterschrift ____________

Coupon ausschneiden, auf Postkarte kleben und einsenden an:
ATARI*magazin*, Softwareversand, Postfach 16 40, 7518 Bretten.

Schlagabtausch

Nicht nur Schach, sondern auch Dame spielt der ST. In beiden Fällen "denkt" er weit voraus.

Nach einigen Schachprogrammen, die inzwischen für den ST auf den Markt gekommen sind, gibt es jetzt auch das beliebte Damespiel in einer entsprechenden Umsetzung. Das Programm läuft voll unter GEM und kann mit einem Monochrom- oder einem Farbmonitor (in der mittleren Auflösung) eingesetzt werden.

16 Bit

Neben der Diskette bekommt der Käufer ein DIN-A5-Ringbuch, das die deutsche Anleitung enthält. Zusätzlich kann man vom Programm aus die Spielregeln aufrufen. Diese setze ich als bekannt voraus, da das Damespiel doch weit verbreitet ist. Anzumerken ist nur, daß man – im Gegensatz zu englischen Programmen dieser Art – mit einer Dame in der Diagonale beliebig viele Felder ziehen kann. Das Blasen oder Pusten eines Steins, der vom Gegner hätte geschlagen werden müssen, wurde beibehalten, obwohl diese Regel bei Turnieren nicht angewendet wird. Dort herrscht absoluter Schlagzwang. Als weitere Besonderheit lassen sich bis zu zehn Züge, die gespeichert werden, rückgängig machen.

Auf dem Monitor präsentiert sich das Programm in einer klaren und übersichtlichen Darstellung. Das Spielbrett wurde in die Bildmitte gesetzt, am oberen Bildschirmrand befindet sich die übliche Menüleiste. Von dort aus können verschiedene Optionen aufgerufen werden. Man hat die Wahl zwischen drei Spielstärken: Anfänger, Amateur und Profi. Auch die Bewegung der Steine kann in drei Stufen geregelt werden.

Die Option Spieler-Menü bietet an:

Mensch-Mensch: Der Computer stellt dann nur das Brett und die Steine zur Verfügung.

Mensch-Computer: Der ST wird zum Spielpartner.

Computer-Computer: Der Rechner spielt alleine. Dies eignet sich gut zur Demonstration.

Da den Regeln entsprechend immer Weiß beginnt, können die Farben auch getauscht werden. Weiterhin besteht die Möglichkeit, eine Partie abzubrechen



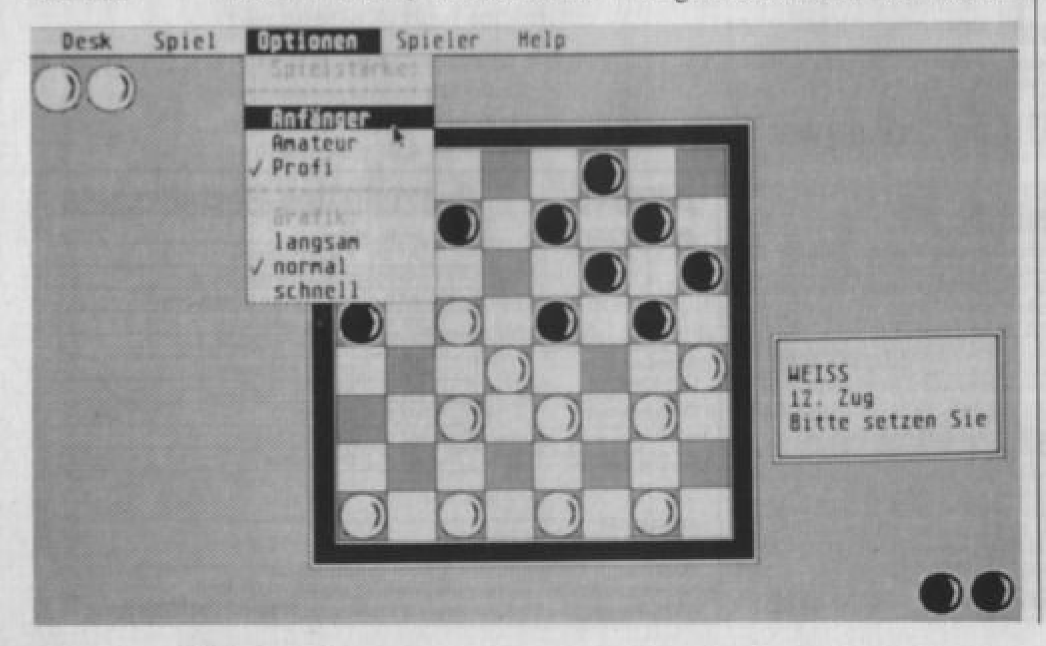
und auf Diskette zu übertragen, um sie später fortzusetzen. Auf Wunsch baut der ST die Grundstellung auf, er akzeptiert aber auch eine vom Spieler entworfene Stellung. Während einer Partie kann man vom Computer einen Zugvorschlag verlangen.

Schon in der Anfängerstufe zeigt der ST eine beachtliche Spielstärke, die sich in den fortgeschrittenen Stufen dank der längeren Rechenzeit noch weiter erhöht. Um erfolgreich zu spielen, muß man sich schon voll konzentrieren. Nachlässigkeiten werden sofort bestraft.

Man kann diese Dame-Umsetzung als gelungen bezeichnen. Vielleicht hätte die grafische Darstellung besser ausfallen können (evtl. wahlweise 2-D- oder 3-D-Darstellung wie bei "Psion Chess"), was aber nur der Optik gedient hätte. Das Spiel selbst ist völlig in Ordnung.

System: Atari ST
Hersteller/Bezugsquelle: Pahlen & Kraus

Rolf Knorre



3 Spielstärken stehen zur Auswahl.

Ein königlicher Monitor

Früher oder später erwacht in jedem Besitzer einer Diskettenstation der Wunsch, mit seinen Disketten mehr anzufangen, als nur einfach Programme zu laden und abzuspeichern. Wenn z.B. das Anfertigen einer Sicherheitskopie unmöglich ist, wäre es doch interessant zu sehen, welcher Kopierschutz dafür zuständig ist. Oder sollen vielleicht eigene Programme geschützt werden? Derlei Beispiele gibt es viele. Ohne ein vernünftiges Werkzeug steht man seinem Laufwerk allerdings recht hilflos gegenüber.

Aus diesem Grund gibt es die sogenannten Diskettenmonitore, die schon unter vielen Namen aufgetaucht sind, eines aber gemeinsam haben: Der Anwender wird in die Lage versetzt, ohne großen Aufwand Disketten zu lesen und zu bearbeiten. Wie weitreichend die Möglichkeiten sind, hängt vom verwendeten Pro-

16 Bit

gramm ab. Eines der komfortabelsten, das ich bisher gesehen habe, ist "Disk Royal". Das liegt sicher auch an der Tatsache, daß es vollständig unter GEM läuft und damit schon von vornherein sehr anwenderfreundlich ist.

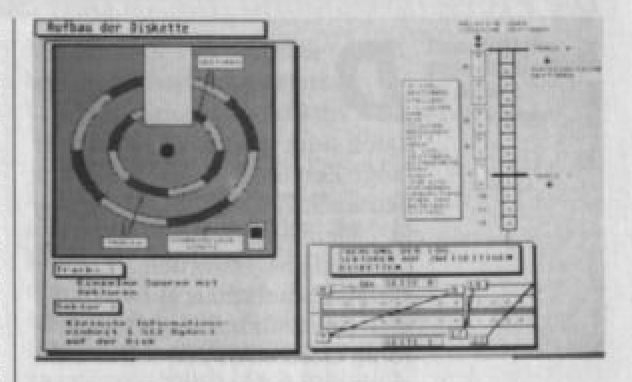
Vorab möchte ich jedoch auf einen Schwachpunkt hinweisen, den alle Programme der Royal-Reihe (wir werden sie nach und nach testen) besitzen: Der Diskette liegt keinerlei Dokumentation auf Papier bei. Natürlich gibt es eine Anleitung. Diese ist aber nur als Dokument auf der Diskette abgespeichert, muß vom Anwender also selbst ausgedruckt oder am Monitor gelesen werden. Der Hersteller rechtfertigt diese Maßnahme mit den hohen Druckkosten, wofür ich kein Verständnis habe. Bei einem Verkaufspreis von 89.– DM ist dies wohl kein schlagkräftiges Argument.

Davon einmal abgesehen, bietet "Disk Royal" kaum Kritikpunkte. Die einzelnen Optionen lassen sich entweder den Pull-Down-Menüs der oberen Bildschirmzeile oder den Symbolleisten am linken und unteren Bildschirmrand entnehmen. Die Bildschirmdarstellung (schwarze Schrift auf weißem Untergrund) kann auf Mausklick invertiert werden. Das Bildzentrum zeigt den gerade zu bearbeitenden Disketteninhalt. Die Bytes und Bits auf einer Diskette können nun nach Spuren, Sektoren und Tracks bearbeitet werden, was sowohl für das Verändern wie auch das Laden und Abspeichern gilt. Die Darstellung erfolgt in hexadezimaler und ASCII-Schreibweise. Neben den Daten eines Files kann man sich z.B. auch eine Cluster-Folge auf dem Bildschirm ansehen.

Weitere Möglichkeiten in Stichworten:

- Kopiere Sektor / Track / Disk / Puffer
- Ändere / Zeige Attribute / Zeit / Datum / Datei-Info
- Einlesen von Boot-Sektor / Directory / FAT 1 & 2
- Drucke Sektoren / Tracks und Hardcopy
- Nächster logischer / physikalischer Sektor

Neben den aufgeführten und vielen weiteren Optionen bietet "Disk Royal" auch drei Hilfsseiten, wie sie besser nicht sein könnten. Besonders der Anfänger wird daraus Nutzen ziehen. Man findet hier in grafischer Form die zum besseren Verständnis der Materie unbedingt erforderlichen Grundlagen des Diskettenbetriebs (z.B. Informationen darüber, wie eine Dis-



Sogar grundlegende Informationen über Diskettenaufbau sind verfügbar.

kette aufgebaut ist und wie sie beschrieben wird).

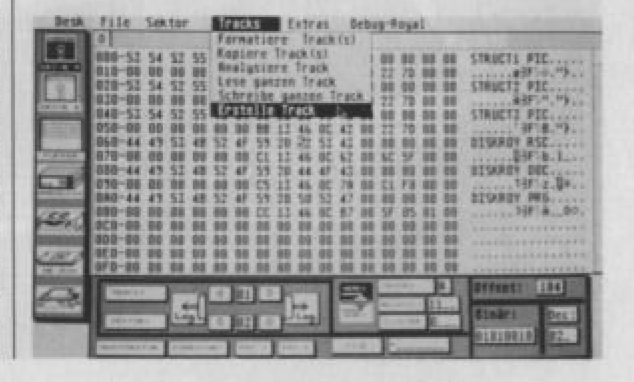
Das Programm beinhaltet außerdem bereits einige Schnittstellen, die auf spätere Erweiterungen abzielen. Das betrifft eine Harddisk, ein CD-ROM und eine Modulerweiterung. Bei ihrem Einsatz bietet der "Disk Royal"-Vertrieb einen kostenlosen Upgrade-Service an, was keineswegs selbstverständlich ist.

Alles in allem ist "Disk Royal" also ein universelles Diskettenwerkzeug, das nebenbei noch mit einem kompletten Kopierprogramm ausgestattet ist. Es eignet sich sowohl für Anfänger als auch für Profis.

System: Atari ST, Monochrom-Monitor
Hersteller: Megasoft
Bezugsquelle: Boston Computer

Stephan König

Zukünftige Technologien bereits vorgesehen: Schnittstelle für CD-Speicher

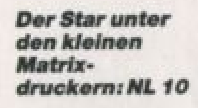


Der Home-Drucker

Die Preisentwicklung macht den Star NL-10 erschwinglich

Der Preis des hier vorgestellten Matrixdruckers ist in den vergangenen Monaten stark gesunken. Kostete er lange Zeit um 1.300 DM, so ist er heute mit 700 bis 800 DM für viele Heimanwender eher erschwinglich geworden. Da er Leistungsmerkmale aufweist, die man sonst nur bei wesentlich teureren Geräten findet, soll er auch entsprechend gewürdigt werden.

Rein äußerlich gleicht der NL-10 seinen Konkurrenten, wenn man von der üppigen Bedienungsleiste an der Frontseite absieht. Diese Minitastatur, vom Hersteller als Panel bezeichnet, ist eine der Besonderheiten, die ihn mit an die Spitze der Drucker für Heimcomputer setzen. Über die verschiedenen Taster, die durch eine Folie geschützt sind, können einige der Möglichkeiten des Geräts eingestellt werden. Dadurch erspart sich der Anwender die manchmal recht umständliche Handhabung der Steuercodes, die auch bei Star Epson-kompatibel gestaltet wurden. Der Panel-Modus bietet folgende Optionen:



Der Star unter den kleinen Matrixdruckern: NL 10

rechten/linken Rand setzen

Papiervorschub vorwärts/rückwärts

Seitenanfang setzen

Selbsttest kurz/lang

Hex-Dump-Modus

mögliche Zeichendarstellung:
80 / 96 / 136 Zeichen/Zeile Standardschrift
80 / 96 / 136 Zeichen/Zeile Doppeldruck
80 / 96 / 136 Zeichen/Zeile kursiv, auch als Doppeldruck möglich, sowie Near Letter Quality in Standard- und Kursivdruck.

Die wichtigsten Funktionen sind sogar beim Druckvorgang noch zu ändern. Der NL-10 bietet natürlich noch weitere Darstellungsmöglichkeiten, die dann aber durch Steuercodes (Escape-Sequenzen) aufgerufen werden (z.B. Unterstreichen, Hoch- und Tiefstellen, Breitschrift, Proportionalschrift usw.). Der NL-10 ist selbstverständlich voll grafikfähig.



Erwähnt werden soll noch eine weitere Besonderheit, die man bei anderen Druckern vermißt. Gemeint ist das Interface zur Verbindung mit dem Computer. Der NL-10 wird ohne fest installiertes Interface gebaut. Der Käufer kann aus verschiedenen Modulen seine Norm auswählen. Zur Zeit gibt es ein Centronics (Parallel)-, ein IBM- und ein Commodore-Interface. Zum Test wurde die Parallel-Schnittstelle verwendet, die an der Geräterückseite in einen Schacht geschoben und mit einer Schraube befestigt wird; dies dauert keine zwei Minuten. Die Schnittstelle bietet dem Anwender ab Werk einen 5-KByte-Speicher. Auch das ist bei anderen Geräten oft nur gegen Aufpreis zu erhalten.

Neben dem Interface-Schacht befindet sich die Leiste mit den DIP-Schaltern. Warum sie nicht nach vorne verlegt wurden, ist unverständlich. Gerade bei einem so professionellen Gerät wie dem NL-10 müßte man dies als komfortablere Lösung erwarten. Allerdings sind die Schalter auch nicht so wichtig, da viel über das Panel geregelt werden kann.

Ein wichtiger Aspekt beim Druckertest ist die Papierverarbeitung. Auch hier ist der NL-10 Spitze. Selbstverständlich können Endlospapier und Einzelblätter verwendet werden. Dabei besteht die Möglichkeit, beide Papiersorten über einen halbautomatischen Einzug einzulegen. Als Option ist auch ein vollautomatischer Einzelblatteinzug zu haben, dann aber nur gegen Aufpreis. Die Papierführung durch Druckwalze oder Stachelräder funktioniert einwandfrei.

Nun zu einem Punkt, der im Heimbereich vielleicht nicht

```
!"#$%&'()*+,-./0123456789:;<=>?§ABCDEFGHIJKLMNOPQRSTUVWXYZÄÖÜ^_'abcdefghijklmno
pqrstuvwxyzäöüß
!"#$%&'()*+,-./0123456789:;<=>?§ABCDEFGHIJKLMNOPQRSTUVWXYZÄÖÜ^_`abcdefghijklmno
pqrstuvwxyzäöüß
!"#$%&'()*+,-./0123456789:;<=>?§ABCDEFGHIJKLMNOPQRSTUVWXYZÄÖÜ^_`abcdefghijklmno
pqrstuvwxyzäöüß
!"#$%&'()*+,-./0123456789:;<=>?§ABCDEFGHIJKLMNOPQRSTUVWXYZÄÖÜ^_'abcdefghijklmno
pqrstuvwxyzäöüß
```

ganz so wichtig ist, aber doch erwähnt werden soll. Es geht um die Druckgeschwindigkeit. Von allen Geräten, die ich bisher getestet habe, ist der NL-10 der schnellste Drucker. Laut Herstellerangabe wird bei Normalschrift eine Geschwindigkeit von 120 Zeichen pro Sekunde erreicht. Der Speedy 100-80 (angeblich 100 Zeichen pro Sekunde) erreicht dies bei weitem nicht. Dabei ist die Geräuschentwicklung fast identisch.

Ein weiteres Beispiel für die Geschwindigkeit ist der Seitenvorschub. Der Star-Drucker jagt das Papier regelrecht heraus und kann bereits wieder eingesetzt werden, während der Speedy immer noch mit dem Seitenvorschub beschäftigt ist. Die Qualität der ausgedruckten Zeichen leidet nicht unter der Geschwindigkeit.

Zusammenfassend kann man sagen, daß beim Star NL-10 einfach alles stimmt. Die robuste Gestaltung des Geräts macht neben den vielen Ansteuerungsmöglichkeiten, der guten Druckqualität und der Schnelligkeit einen sehr guten Eindruck. Auch bei sehr langen Ausdrucken arbeitet der NL-10 einwandfrei.

Beachtliche Schriftqualität in dieser Preisklasse

Wer sich heute einen Drucker anschaffen will, sollte sich lange überlegen, ein auf den ersten Blick preiswerteres Gerät oder den NL-10 zu kaufen. Ein Preisunterschied von vielleicht 200 DM ist sicher für manchen Anwender erheblich, doch zahlt sich dies aus, wenn man nur an das auswechselbare Interface denkt.

Rolf Knorre

SERIE

Schnelle Umwege in Action!

Im zweiten Action!-Center behandeln wir die Interrupts für flotte Programme.

Die Benutzung von Interrupts ist ein Kapitel, das normalerweise nur Assembler-Programmierern vorbehalten ist. Gerade ihre Beherrschung stellt jedoch ein sehr gutes Hilfsmittel für eine Programmiersprache dar, da durch einen periodisch wiederkehrenden Interrupt (beim Atari VBI genannt) sogar eine Art von Parallelverarbeitung möglich ist. Das bedeutet nicht mehr und nicht weniger, als daß zwei Programme (quasi) gleichzeitig bearbeitet werden. Sie können sich vorstellen, daß es mit diesen Fähigkeiten kein Problem ist, fließende Animation mit Musik oder Geräuschen zu verbinden.

Nun wäre es aber schade, wenn eine so vielversprechende Fähigkeit nur in Assembler nutzbar sein sollte, doch ist es ganz einfach eine Frage der Rechenzeit. Der bereits erwähnte VBI wird von der Hardware des Rechners 50mal pro Sekunde aufgerufen. Dann verzweigt der Programmablauf zu einem speziellen Programmteil im ROM des Computers und kann, wie wir später noch sehen werden, auch auf eigene Programme umgelenkt werden. Es ist klar, daß die

8 Bit

vom VBI aufgerufene Routine auch innerhalb von 20 Millisekunden (1/50 sec) abgeschlossen sein muß, denn sonst würde sie nochmals aufgerufen, bevor sie beendet ist. Ein heilloses Durcheinander wäre die sichere Folge.

Eine langsame Sprache wie das Atari-Basic könnte in dieser Zeit nur sehr wenige Befehle abarbeiten. (Ein Befehl wie A=B*C nimmt alleine schon ca. 12 Millisekunden in Anspruch!) Ein Assembler-Programm kann dagegen in 20 Millisekunden eine ganze Reihe von Befehlen bearbeiten. Während dieser Zeitspanne stehen etwa 23.000 bis 33.000 Taktzyklen zur Verfügung, die für einige tausend Maschinenbefehle ausreichen. Damit läßt sich schon etwas anfangen.

Wer die letzte Folge des Action!-Centers verfolgt hat, der weiß, daß diese Sprache einem Assembler-Programm in Sachen Geschwindigkeit nicht viel nachsteht. Warum sollte man also Interrupt-Routinen nicht in Action! codieren?

Im Prinzip ist das kein Problem, nur eine Schwierigkeit gibt es dennoch zu überwinden. Jede Sprache benötigt einige Hilfsregister zur Ablage von Zwischenergebnissen. Es darf keinesfalls geschehen, daß die Interrupt-Routine die Hilfsregister des parallel laufenden Programms (man spricht hier auch vom Vordergrundprogramm) überschreibt. In Assembler betrifft dies nur die internen Register des Prozessors wie Akku und Index-Register, die man zu Beginn auf den Stapel rettet und vor dem Ende der Interrupt-Routine wieder zurückholt

Action! als Compilersprache benötigt da schon ein paar Register mehr. Grundsätzlich geht man aber genauso vor: Zu Beginn des Interrupts rettet man alle internen Register auf den Stapel und holt diese vor Ende des Interrupts auf dem umgekehrten Weg wieder zurück. Diese Auf-

> **„Strenge Trennung der Datenbereiche"**

gaben überträgt man zwei Code-Blocks, die im Beispielprogramm (Listing 1) am Anfang und Ende der Prozedur Vbipgm () definiert sind.

Der erste Block überträgt einen Teil der Zero-Page auf den Stack, hauptsächlich die Bereiche von $ 80 bis $ 8F, die Action! zur Berechnung von Formeln benützt, sowie von $ A0 bis $ AF, einen Bereich, der vorwiegend zur Parameterübergabe Verwendung findet. Daneben werden noch vereinzelte Speicherzellen und natürlich die Prozessorregister auf den Stack geschoben. Der Code-Block am Ende von Vbipgm () kehrt diesen Vorgang um und stellt somit den ursprünglichen Zustand wieder her. Das Vordergrundprogramm merkt somit von der Unterbrechung nichts, was schließlich auch der Zweck der Übung war.

Bei der Programmierung der Interrupt-Routine muß man streng auf eine Trennung der Datenbereiche von Interrupt zu Vordergrund achten. Action! erleichtert dies durch die Verwendung von lokalen Variablen; dagegen sind globale mit Vorsicht zu genießen, können jedoch zur Kommunikation zwischen den

Interrupts in Action!

```
;**************************************
;VERTICAL-BLANK INTERRUPT IN ACTION!
;
;P.FINZEL                         1987
;**************************************

DEFINE  JMP     ="$4C",
        XITVBV  ="$E462"

;--------------------------------------
;Hardware- & Schattenregister
;--------------------------------------
CARD Vvblkd=$0224, Vvblki=$0222

BYTE
Dmactl=$D400, Prior =$26F,
Nmien =$D40E, Colbk =$D01A,
Gractl=$D01D, Pmbase=$D407,
Crsinh=$02F0, SDmctl=$022F

BYTE ARRAY
Color(5)=$02C4, PColr(4)=$02C0,
Hposp(4)=$D000, Sizep(4)=$D008,
Trig(4) =$D010, Grafp(4)=$D00D

;--------------------------------------
;Graphik-Daten
;--------------------------------------
BYTE ARRAY Shape(0)=
[ $80 $F0 $60 $50 $08 $04 $02 $01 ]

CARD ARRAY PMAdr(4)=[0 0 0 0]

BYTE ZeigerX, ZeigerY

;--------------------------------------
;PM-Graphik einschalten
;--------------------------------------
PROC Pmgraphics(BYTE Basis)
BYTE i
CARD Adr
BYTE Adrhi=Adr+1

Pmbase=Basis
Adr    =0
Adrhi  =Basis+4
Gractl=3
SDMCTL=$3A
FOR i=0 TO 3
    DO
    Zero(Adr,256)
    Pmadr(i)=Adr
    Adr     ==+256
    Sizep(i)=0
OD
RETURN

;--------------------------------------
;Einbinden von Interrupt-Routinen
;--------------------------------------
PROC Setvbi(BYTE POINTER Vektor)
Nmien=0 Vvblkd=Vektor Nmien=$40
RETURN

;--------------------------------------
;VBI-Routine
;--------------------------------------
PROC Vbipgm=*()

;zuerst Register retten
[ $A2   7 $B5 $C0 $48 $B5 $A8
  $48 $B5 $A0 $48 $B5 $80 $48
  $CA $10 $F1 ]

Zero(PMAdr(0)+ZeigerY,8)
IF (Stick(0)&1)=0 THEN
      IF ZeigerY>32 THEN
             ZeigerY==-1
      FI
ELSEIF (Stick(0)&2)=0 THEN
      IF ZeigerY<224 THEN
             ZeigerY==+1
      FI
FI
IF (Stick(0)&4)=0 THEN
      IF ZeigerX>48 THEN
             ZeigerX==-1
      FI
ELSEIF (Stick(0)&8)=0 THEN
      IF ZeigerX<208 THEN
             ZeigerX==+1
      FI
FI
MoveBlock(PMAdr(0)+ZeigerY,Shape,8)
Hposp(0)=ZeigerX

;Register holen...
[ $A2 0 $68 $95 $80 $68 $95 $A0 $68
  $95 $A8 $68 $95 $C0 $E8 $E0 8 $D0
  $EF ]
;...und VBI beenden
[JMP XITVBV]


;--------------------------------------
;Das Hauptprogramm
;--------------------------------------
PROC HAUPT()

  Pmgraphics($80) ;PMG ab Adr. $8000
  Prior=1         ;Prioritaet
  PColr(0)=$CE    ;Farbe Player
  ZeigerX=120     ;Startpunkt
  ZeigerY=120
  SetVBI(VBIPgm)  ;Interrupt
PUT($7D)          ;Bildschirm loeschen
Crsinh=1          ;Cursor aus
POSITION(10,8)    ;Kasten zeichnen
Put(17) Put(18) Put(18) Put(5)
POSITION(10,9)
Put(124) Put(79) Put(75) Put(124)
POSITION(10,10)
Put(26) Put(18) Put(18) Put(3)
DO
IF TRIG(0)=0 THEN
  IF (ZEIGERX>88) AND
     (ZEIGERX<104) THEN
     IF (ZEIGERY>96) AND
        (ZEIGERY<120) THEN
          EXIT
     FI
  FI
FI
OD
SETVBI(XITVBV)
RETURN
```

Programmteilen eingesetzt werden.

Das Einbinden einer Action!-Prozedur in den VBI ist mit der Routine SETVBI () sehr leicht zu erledigen. Als einzigen Parameter muß man der Routine die Adresse des VBI-Programms als CARD übergeben. Dort wird zunächst der VBI mittels des Registers NMIEN abgeschaltet, dann der Vektor VVBLKD auf die gewünschte Adresse geändert und schließlich der Interrupt per NMIEN wieder zugelassen. Der Aufruf SETVBI (VBIPGM) hat somit zur Folge, daß die Prozedur VBIPGM 50mal pro Sekunde aufgerufen wird.

Interrupts sind Unterbrechungen des Programmablaufs für bestimmte Aufgaben. Macht man sich diese zunutze, so ist eine Art Parallelverarbeitung programmierbar.

Das Interrupt-Programm VBIPGM selbst beginnt, wie wir schon gesehen haben, mit dem Code-Block zum Retten der Register. Daran schließen sich die Befehle an, die im VBI ausgeführt werden sollen. Hier kann ganz normal in Action! programmiert werden; auch andere Prozeduren lassen sich selbstverständlich aufrufen. Vorsicht ist lediglich bei Befehlen geboten, die I/O-Funktionen aufrufen (z.B. OPEN, PRINT usw.) oder das Betriebssystem benötigen (PLOT, DRAW usw.). Den Abschluß bilden der Code-Block zur Wiederherstellung der Registerinhalte und ein Sprung zum Label XITVBV ($E462), der wiederum als Code-Block erfolgt.

Das Beispiel im Listing benutzt nun diese Möglichkeiten, um eine Art Mauszeiger mit dem Joystick bewegen zu können. Der Zeiger ist nichts weiter als ein Player, der sich durch den Einsatz des Interrupts vollkommen fließend und unabhängig vom Vordergrundprogramm bewegen läßt. Um das Ganze nicht allzu sehr aufzublähen, wird im Vordergrund nur eine sehr einfache Funktion erledigt, nämlich gewartet, bis der Zeiger auf das Kästchen OK deutet und der Knopf am Joystick gedrückt wird. Sie sehen, mit dieser Technik läßt sich eine mausähnliche Bedienung sehr leicht erzeugen.

Beachten Sie bitte, daß Bewegung des Zeigers und Abfrage der Position unabhängig voneinander erledigt werden. Als Verbindung zwischen den Programmteilen dienen nur die globalen Variablen Zeiger _X und Zeiger _Y.

Mit diesem Grundkonzept kann man aber auch sehr viel kompliziertere Programmsysteme aufbauen. Denkbar wäre z.B. ein Spiel, das alle Bewegungen der Players oder ein eventuelles Scrolling im VBI durchführt, während die Logik des Spiels (Ablauf, Punktezählung usw.) im Vordergrund erledigt wird. Warum nicht umgekehrt? Das liegt an der Natur des VBI, der bekanntlich in der vertikalen Austastlücke des vom Computer erzeugten Videobildes ausgelöst wird. Alle Grafikänderungen, die im VBI stattfinden, sind daher nicht mit Störungen (z.B. Flimmern) behaftet.

Interessant ist noch die Prozedur PMGraphics (), mit der die PM-Grafik eingeschaltet wird. Die Anfangsadressen des Videospeichers der einzelnen Players sind danach im CARD-Array PMAdr zu finden. Die Bewegung des Players geschieht in der Routine..., indem zuerst der Zeiger an der alten Position gelöscht, danach der Joystick abgefragt und schließlich die Form des Zeigers aus dem Array SHAPE an die neue Position kopiert wird. Dank der enormen Geschwindigkeit eines Action!-Programms nützt die Routine den Zeitrahmen des VBI nur zu einem geringen Teil aus; tatsächlich sind bei geschickter Programmierung sehr viel aufwendigere Routinen möglich.

„Enorme Geschwindigkeit"

Man sollte natürlich auch bedenken, daß die Rechenzeit des VBI von der gesamt verfügbaren abgeht, d.h., bei einer sehr langen Routine kommt das Vordergrundprogramm fast zum Stillstand.

Auf ähnliche Weise lassen sich auch die anderen System-Interrupts (DLI oder I/O-Interrupts) auf Action!-Programme umlenken, doch mehr davon in einer der nächsten Folgen. Sie sehen, daß man mit dieser C-ähnlichen Sprache auch so komplizierte Sachverhalte wie Interrupts mühelos in den Griff bekommt, ohne auf die langwierige Assembler-Programmierung ausweichen zu müssen. Es ist daher kein Wunder, wenn die Betriebssysteme der neuen 16-Bit-Generation (z.B. ST oder Amiga) ausschließlich in C programmiert sind.

Peter Finzel



Meister der Disketten

Benötigen Sie einen professionellen Programmierschutz für Ihre Software? Kein Problem mit dem neuen Programm "Diskmaster", das wir Ihnen hier vorstellen.

Ohne Zweifel übt das Thema Kopierschutz auf viele Leute eine enorme Faszination aus. Mit dem Programm "Diskmaster" ist es für jedermann möglich, einen Kopierschutz zu erzeugen, der selbst professionellen Ansprüchen genügt. Voraussetzung ist ein 1050-Laufwerk mit einer Happy-Erweiterung bzw. ein dazu kompatibles Produkt.

Erstaunlicherweise kann "Diskmaster" mit diesem Hardware-Zusatz sogar Kopierschutzformate erstellen, die sich nicht einmal mit der Happy-Erweiterung kopieren lassen, geschweige denn mit einem normalen Sektorkopierer.

8 Bit

Der zentrale Teil des Programms ist der sogenannte Format-Editor, mit dem ein Kopierschutzformat in einer Art eigener Programmiersprache entworfen werden kann. Dabei läßt sich jedes einzelne Byte des Formats bestimmen; mehr oder weniger kann der FDC (der Floppy-Controller der 1050-Station) direkt angesprochen werden. Wer sich schon einmal mit der Floppy-Anschaltung beim Atari beschäftigt hat, weiß, daß dies keine einfache Sache ist. Der FDC gehorcht nämlich nicht den Befehlen des Mikroprozessors im Computer, sondern verfügt über eine eigene CPU, die lediglich über einen seriellen Bus mit der zentralen CPU in Verbindung steht. Zudem ist das normale Betriebsprogramm der Floppy-CPU penibel dazu ausgelegt, solche nicht vorgesehenen Aktionen strengstens zu verhindern.

Durch den Format-Editor lassen sich Kopierschutzformate mit doppelten Sektoren oder absichtlichen Lesefehlern erzeugen oder einfach mehr Sektoren als gewöhnlich auf einer Spur unterbringen. Nach Wunsch kann dabei der Single- oder Double-Density-Modus verwendet werden. Dennoch ist es nicht ganz leicht, einen Kopierschutz zu erstellen, da man schließlich jedes Byte auf der Diskette berücksichtigen muß. Glücklicherweise enthält die Rückseite der Diskette noch eine Reihe fertiger Formate als Files, die man sofort ausprobieren und nachvollziehen kann. Außerdem sind auch einige Basic- und Assembler-Programme zu finden, welche die Abfrage der verschiedenen Kopierschutzformate als Beispiel zeigen.

Neben dem Editor besitzt "Diskmaster" noch einige weitere Funktionen, die zum Speichern und Laden von Formaten und sonstigen Daten dienen. Natürlich darf auch eine Option zum Aufbringen des Formats nicht fehlen. Erwähnenswert ist noch die Funktion SPUR EINLESEN; sie überträgt eine ganze Spur mit allen darauf befindlichen Daten, also auch Sektor-Header und Prüfsummen-Bytes, in den Speicher des Computers.

"Happy-kompatible Erweiterung erforderlich"

Die sehr ausführliche Dokumentation umfaßt 15 DIN-A4-Seiten. Sie ist – was ich für besonders wichtig halte – nicht nur auf die Bedienung des Programms zugeschnitten, sondern erklärt auch einige wichtige Grundlagen der internen Funktion der Diskettenstation. Das ist gerade bei der Erzeugung eines Kopierschutzes von großer Bedeutung, da zu diesem Zweck alle Eigenheiten der Floppy genutzt werden. Allerdings ist es natürlich unmöglich, eine so komplexe Thematik dem Laien auf wenigen Seiten erschöpfend zu erklären; einige Vorkenntnisse sind hier schon erforderlich. Wer mit Begriffen wie Gapbyte und CRC-Generator nichts anzufangen weiß, wird bei der Lektüre schnell überfordert sein.

Trotzdem ist "Diskmaster" mit seinem günstigen Preis von 24.90 DM allen zu empfehlen, die mehr über ihr Diskettenlaufwerk und seine Funktionsweise wissen möchten. Beim Kauf ist unbedingt zu beachten, daß man zur Anwendung des Programms zumindest eine Happy-kompatible Erweiterung braucht. Funktionsähnliche Produkte wie Speedy 1050 oder Turbo 1050 sind hier nicht verwendbar. Im Zweifel fragen Sie am besten Ihren Händler.

Zu haben ist das Programm beim Software-Versand des **ATARI***magazins* (Bestellschein S. 29).

Peter Finzel

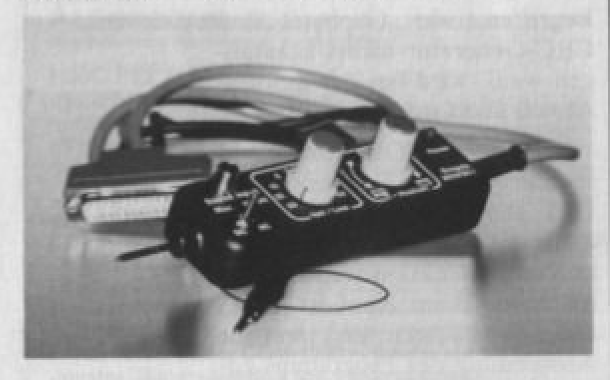
Der ST-Oszillograph

Ein preiswertes Angebot aus Hard- und Software für die Messungen in der Elektronik.

Das Herz aller Elektroniker, die über einen Atari ST verfügen, dürfte nach diesem Bericht höher schlagen. Das uns vom Microcomputerlabor zur Verfügung gestellte Hard- und Softwarepaket "Oszillograph" ist nämlich eine Komplettlösung, die in Preis und Leistung vorerst unschlagbar sein könnte.

Geliefert wird das Paket in einem kleinen Kunststoffkoffer, der der sicheren Verpackung dient, aber auch als Transportmittel eingesetzt werden kann (fehlt eigentlich nur noch der passende Koffer für den ST). Darin befindet sich – verpackt in viel Schaumstoff – das Meßgerät mit Tastspitze, Masseklemme, zwei Potentiometern, einem Schalter, dem Sound-Eingang und einer Power-Kontrolldiode, alles fest verbunden mit einem ca. 1,70 m langen Kabel. Außerdem findet man eine deutsche Anleitung und zwei Disketten mit der Software (eine davon dient allerdings nur Demozwekken).

Mit diesem Hardware-Zusatz wird der ST zur Elektronik-Meßstation



Verbunden wird das Kabel computerseitig mit Drucker- und Joystickport; weitere Installationsarbeiten sind nicht notwendig. Von der Diskette kann nun die Software geladen werden. Im Prinzip stehen dann ein digitales Oszilloskop, ein Speicheroszilloskop und ein Oszillograph zur Verfügung. Man kann also periodische Spannungsänderungen (z.B. Sinus oder Rechteck) oder auch unperiodische und extrem langsame Spannungsänderungen wie das Prellen eines Relais, die Entladekurve von Akkus oder das Regelverhalten von Heizungsanlagen sichtbar machen, und das nicht nur als Meßwert, sondern auch in grafischer Form auf Monitor oder Drucker.

Somit bietet sich eine Vielzahl von Einsatzmöglichkeiten, die teilweise aber nur mit zusätzlicher Soft- und Hardware realisiert werden können. Der Hersteller gibt dazu einige Beispiele:

Automatische EKG-Auswerter

Sprachanalysator

Nachhallmeßgerät

Intelligentes Überwachungssystem

Frequenzanalysator

Sound Sampler

Auch ohne zusätzliche Investitionen ist der "ST-Oszillograph" aber voll einsatzfähig und ermöglicht Anwendungen, die sonst nur mit erheblich teureren Geräten machbar wären.

Die Bedienung ist relativ einfach, da die Software voll mausgesteuert ist. So erfolgt die Umschaltung zwischen den Hauptbetriebsarten Oszilloskop und Speicheroszilloskop z.B. durch einfachen Mausklick auf die entsprechende Option. Der Bildschirmaufbau ist dem abgedruckten Foto zu entnehmen. Etwas schwach ausgefallen ist lediglich die Anleitung, die sich auf ein Minimum an Informationen beschränkt. Dem Fachmann wird dies aber kaum Schwierigkeiten bereiten.

Nun noch einige technische Daten zur Information:

Die Bildschirmdarstellung besteht aus 8×10 Linien mit einem Speicher von 50 Seiten.

Y-Verstärker: 0,1 – 0,2 – 0,5 – 1,0 – 2,0 – 5,0 – 10 – 20 – 50 Volt/Linie, kalibriert.

Timebase: 50 µsec/Linie bis 500 sec/Linie, kalibriert.

Meßgeschwindigkeit: max. 60.000 Messungen pro Sekunde.

Meßdauer: zwischen 1 msec bis max. 69,5 Stunden.

Auf der Diskette befindet sich ein weiteres Programm, das die Realisierung eines Sound-Samplers erlaubt. Wer das Spiel "Starglider" kennt, weiß, was mit einem solchen Gerät möglich ist. Vereinfacht ausgedrückt, kann man mit dem "Sampler" (oder auch Sound-Digitizer) Musik oder Sprache digitalisieren und in den ST überspielen. Hierzu besitzt das Meßgerät eine Chinchbuchse, an die z.B. ein Cassettenrecorder angeschlossen werden kann.

Nach Laden der Software kann der Digitizer sofort eingesetzt werden. Der Hersteller weist darauf hin, daß diese Anwendung nur ein Nebenprodukt ist und nicht mit reinen Digitizern zu vergleichen ist. Doch sollte er nicht so tiefstapeln; die Möglichkeiten, die der "Sound Sampler" bietet, sind schon bemerkenswert. Hierzu muß man allerdings etwas experimentieren, da mit der Minianleitung auf der Diskette kaum etwas anzufangen ist.

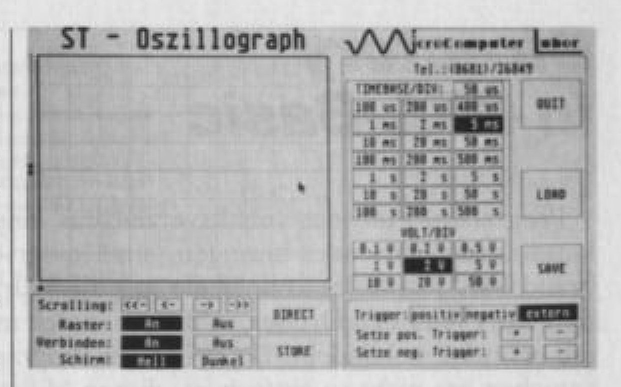
Die Länge des zu digitalisierenden Sounds richtet sich nach der Speicherkapazität (beim Test wurde ein Megabyte ST verwendet) und der Abtastrate, die per Software eingestellt werden kann. Die Bandbreite liegt zwi-

schen 2,5 und 45 kHz. Um Musik zu verarbeiten, hat sich die Einstellung 12,5 kHz als optimal erwiesen. Geht man in den tiefen Bereich um 5 kHz, sollte ein Tiefpaßfilter vorgeschaltet werden, da andernfalls sogenannte Alias-Effekte auftreten, die bei der Wiedergabe extreme Verzerrungen hervorrufen. Eine hohe Abtastrate von z.B. 45 kHz erhöht zwar die Klangqualität, bedeutet aber in der Praxis, daß nur wenige Sekunden Sound in den Speicher passen. Da der Monitorlautsprecher einen wesentlich geringeren Frequenzgang hat, ist die Einstellung 12,5 kHz völlig ausreichend. In diesem Bereich kann man rund 1 bis 2 Minuten Musik digitalisieren.

Zuvor sollte über die Option DIRECT eine Aussteuerung erfolgen, um Verzerrungen zu vermeiden. Danach wird einfach RECORD angeklickt und der Recorder gestartet. Ist der Speicher voll, erscheint eine entsprechende Meldung. Den digitalisierten Sound kann man nun beliebig oft abhören. Je nach verwendeter Musikart ist die Klangqualität erstaunlich gut.

Der Speicherinhalt kann auch weiterbearbeitet werden. Mit der Option MARK lassen sich z.B. Blöcke markieren, danach an andere Stellen kopieren, vervielfältigen oder spiegeln. Ein solcher Block kann beliebig lang sein, aber auch nur Bruchteile einer Sekunde betragen. Selbstverständlich kann man den Speicherinhalt auch auf Diskette übertragen und später wieder verwenden.

Der "ST-Oszillograph" steht mit einem enormen Preis/Leistungs-Verhältnis fast konkurrenzlos da. Durch die vielfältigen



Einsatzmöglichkeiten und die mitgelieferte Software wird der positive Eindruck nur noch verstärkt. Ob als Meßgerät oder als "Sound Sampler", beide Anwendungen haben den Test einwandfrei überstanden.

Ein anwenderfreundliches Programm ergänzt die Hardware.

System: Atari ST mit Farb- oder S/W-Monitor
Hersteller/Bezugsquelle: Microcomputerlabor, Saarbrükken

Rolf Knorre

Directory in GFA-Basic

Programme, die das Inhaltsverzeichnis einer Diskette ausdrucken oder anzeigen, sind leider oft unbrauchbar. Entweder können sie nur die Wurzel der Directory oder den Ordner, in dem man sich momentan befindet, behandeln. Da es in allen Programmiersprachen gar nicht so einfach ist, diesen Mißstand zu beheben, haben wir den "Director" in GFA-Basic realisiert.

Das Programm gibt die gesamte Directory einer Diskette mit allen Ordnern und Unterordnern auf dem Bildschirm aus. Die Daten können ausgedruckt, gespeichert oder weiterverarbeitet werden. Dies bildet dann den Grundstock zu einer eigenen Diskettenverwaltung.

Hier nun das Grundprogramm zur individuellen Lösung. Damit ist es möglich, von der Wurzel aus in immer tiefere Gefilde des File-Suchbaums vorzudringen und alle Dateien (dies können auch Programme sein!) zu finden. Um sich die vielen möglichen Pfadnamen zu merken, richten wir einen Stack ein, der auf 100 dieser Bezeichnungen dimensioniert ist. Der Pfadname stellt einen sogenannten roten Faden auf der Diskette dar, damit das Betriebssystem weiß, wo sich eine Datei befindet.

Prozeduren

Zur Prozedur "Directory" gibt es noch einiges zu sagen. Interessant ist der Joker$, in dem die Maske des Suchobjekts vereinbart wird. Hiermit ist es auch möglich, durch geeignete Extender-Wahl nur bestimmte Dateitypen zu finden (Joker$="*.BAS" findet z.B. alle Basic-Programme). Bei der GFA-Version 1.0 wird man am Wort Void keine Freude haben, da der Interpreter hier streikt. Doch keine Sorge; Void bedeutet nur, daß die Rückgabeparameter der Funktion nicht gebraucht werden. Statt void gemdos (........) z.B. einfach xx=gemdos (78, L: varptr(Joker$), 8) schreiben. Hier haben wir die 8 als Diskettennamenskennung gewählt.

Eigentlich ist es etwas ungewöhnlich, daß der GEMDOS-Aufruf SFIRST als Erfolgsparameter eine 0 zurückgibt. Aber es müssen ja verschiedene Fehlerarten berücksichtigt werden, die für unsere Anwendung jedoch nicht zum Tragen kommen.

Die Prozedur "Datei_auslesen" liest, wie ihr Name schon sagt, sämtliche Dateien aus dem aktuellen Ordner aus. Dabei wird mit dem Attribut getestet, ob eine Datei oder ein Ordnereintrag vorliegt. Bei Attribut=16 handelt es sich um einen Ordner, der auf dem Stack landet.

Die Prozedur "Dta_auslesen" ist das Kernstück des Programms. Voraussetzung zum Verständnis ist der Aufbau des DTA-Buffers (Disk Transfer Adress Buffer). Er besteht aus 41 Byte, wobei die ersten 21 für unseren Zweck nicht interessant sind. Zuerst kommt an der 22. Stelle das Attribut mit einem Byte:

- 0 = lesen/schreiben
- 1 = schreibgeschützt
- 2 = verborgener Eintrag
- 4 = verborgener Systemeintrag
- 8 = Diskettenname
- 16 = Unterordner
- 32 = Datei wurde beschrieben und ordnungsgemäß geschlossen

Uhrzeit und Datum

Als nächstes folgt 1 Wort, das die Uhrzeit enthält:

Bit 0-4 Sekunden (in Zweierschritten)
Bit 5-10 Minuten
Bit 11-15 Stunden

An der 24. Stelle steht das Wort mit dem Datum:

Bit 0-4 Tag
Bit 5-8 Monat
Bit 9-15 Jahr (ab 1980)

Die Länge folgt an der 26. Stelle mit einem Langwort.

Dann schließt sich eine 12-Byte-Zeichenkette mit dem von SFIRST gefundenen Dateinamen an, der mit einem 0-Byte abgeschlossen wird. Den Inhalt des DTA-Buffers übertragen wir in die entsprechenden Strings und Masken.

Norbert Schmantz / Herbert Polzer

Director

```
' ------------------------------ D I R E C T O R ------------------------------
' --- Diese Routine liefert das komplette Inhaltsverzeichnis einer Diskette ---
' ---              von Ing.Norbert Schmautz & Herbert Polzer           + TT86 ---
' ---              - ein Megabrain-Produkt des Allgäu-Teams -                ---
' ------------------------------------------------------------------------------
Anzahl=300
Dim Datei$(Anzahl),Laenge$(Anzahl),Uhr$(Anzahl),Datum$(Anzahl),Ordner$(Anzahl)
Dim Stack$(100)
Laufwerk$="A"
Datei=0
'
Gosub Directory(Laufwerk$)
'
If Datei>0
  Print Diskname$
  For L=1 To Datei
```

```
    Print Datei$(L);" ";Laenge$(L);" ";Uhr$(L);" ";Datum$(L);" ";Ordner$(L)
  Next L
Endif
'
End
'
'
'
Procedure Directory(Laufwerk$)
  Local A,L,Dta,Stack,Attribut,Fehler
  Local Joker$,Datei$,Laenge$,Uhr$,Datum$,Ordner1$,Ordner2$
  Joker$="*.*"                                ! Dateisuchname
  Dta=Gemdos(47)                              ! DTA-Adresse ermitteln
  Poke Dta+30,0                               ! DTA Name loeschen
  Chdrive (Asc(Laufwerk$)-64)                 ! Laufwerk setzen
  Chdir "\"                                   ! Directory Wurzel setzen
  Void Gemdos(78,L:Varptr(Joker$),8)          ! Diskettenname ermitteln (Fsfirst)
  Gosub Dta_auslesen
  ' -------------------------- Diskettenname und freier Diskettenspeicher -----
  Diskname$=Datei$+" mit  "+Str$(Dfree(0))+" Bytes frei"
  Do
    Gosub Datei_auslesen
    Exit If Stack=0                           ! Alle Ordner durchsucht
    Chdir Stack$(Stack)                       ! Naechsten Ordner setzen
  Loop
Return
'
Procedure Datei_auslesen
  Ordner1$=String$(30," ")
  Ordner2$=Stack$(Stack)
  Lset Ordner1$=Ordner2$
  If Stack>0
    Dec Stack
  Endif
  Fehler=Gemdos(78,L:Varptr(Joker$),1+2+4+16+32) ! Dateien ermitteln (Fsfirst)
  If Fehler=0                                    ! Diskette ok ?
    Gosub Dta_auslesen
    Do
      If Attribut=16                             ! Ordner
        If Right$(Datei$)<>"."                   ! Directoryverweise ?
          Inc Stack
          Stack$(Stack)=Ordner2$+"\"+Datei$
        Endif
      Else                                       ! Datei
        Inc Datei
        While Len(Datei$)<12
          Datei$=Datei$+" "
        Wend
        Datei$(Datei)=Datei$
        Laenge$(Datei)=Laenge$
        Uhr$(Datei)=Uhr$
        Datum$(Datei)=Datum$
        Ordner$(Datei)=Ordner1$
      Endif
      Fehler=Gemdos(79)                          ! Naechste Datei holen (Fsnext)
      Exit If Fehler<>0                          ! Directoryende erreicht
      Gosub Dta_auslesen
    Loop
  Endif
Return
'
Procedure Dta_auslesen
  ' -------------------------- D a t e i n a m e ------------------------------
  Datei$=""
  For L=1 To 14
    A=Peek(Dta+29+L)
    Exit If A=0                                  ! Ende von Dateiname erreicht
    Datei$=Datei$+Chr$(A)
  Next L
  ' -------------------------- L a e n g e ------------------------------------
  Laenge$="        "
  Rset Laenge$=Str$(Lpeek(Dta+26))
  ' -------------------------- U h r ------------------------------------------
  A=Dpeek(Dta+22)
  Uhr$="00:00:00 "
  H$=Str$((A And &X1111100000000000)/2048)
  Mid$(Uhr$,3-Len(H$))=H$
  H$=Str$((A And &X11111100000)/32)
  Mid$(Uhr$,6-Len(H$))=H$
  H$=Str$((A And &X11111)*2)
  Mid$(Uhr$,9-Len(H$))=H$
  ' -------------------------- D A T U M --------------------------------------
  A=Dpeek(Dta+24)
  Datum$="00.00."
  H$=Str$(A And &X11111)                         ! Tag
  Mid$(Datum$,3-Len(H$))=H$
  H$=Str$((A And &X111100000)/32)                ! Monat
  Mid$(Datum$,6-Len(H$))=H$
  Datum$=Datum$+Str$(((A And &X1111111000000000)/512)+1980) ! Jahr
  ' -------------------------- A T T R I B U T --------------------------------
  Attribut=Peek(Dta+21)
Return
```

ST-Programm zur Dateiverschlüsselung

Oftmals ist es wünschenswert, bestimmte Dateien vor fremder, unbefugter Einsichtnahme zu schützen. Im folgenden werden wir ein Programm kennenlernen, das es erlaubt, Files zu verschlüsseln. Nur wer die Verschlüsselungsparameter (ein Wort und eine Zahl) kennt, ist in der Lage, an den Originalinhalt einer solchen Datei heranzukommen. So ist sichergestellt, daß ein Diskettendieb nichts mit den Daten anfangen kann, die er gestohlen hat, oder daß mehrere Personen mit denselben Disketten arbeiten können, auch wenn der eine keinen Einblick in die Arbeit des anderen bekommen soll.

Zum Programm

Das Programm ist in Assembler geschrieben und daher ausgesprochen schnell. Das Ver- oder Entschlüsseln einer 30 KByte langen Datei dauert z.B. etwa fünf Sekunden (plus Zeit für Diskettenzugriffe). Die Handhabung des übersetzten Programms CRYPTO .TOS ist einfach. Unter dem Menüpunkt Dateiname gibt man den Namen des zu bearbeitenden Files an.

Hier ist auch ein Suchausdruck erlaubt, also ein String, der Fragezeichen und Sternchen enthalten darf. Die Dateimaske bestimmt beim Verschlüsseln den Namen der neu erzeugten Datei. Ein Sternchen in der Dateimaske wird durch den Namen der Originaldatei ohne Extension ersetzt. Dieser (ohne Dateibaumpfad) wird mit in das neue File übertragen. Deshalb ist beim Entschlüsseln nicht die Dateimaske für den Namen der neu kreierten Datei verantwortlich, sondern es wird ein File mit dem Originalnamen angelegt, allerdings in der Directory, aus der CRYPTO .TOS aufgerufen wurde.

Wie funktioniert die Verschlüsselung?

Zur Anwendung kommt ein im Grunde recht einfaches Verfahren, das auf byteweiser Exklusiv-Oder-Verknüpfung (EXOR) des Dateiinhalts mit einem Schlüssel beruht, der aus Verschlüsselungswort und -zahl gewonnen wird. Das Wort wird in zwei Teile aufgespalten, wobei der hintere ein Zeichen länger ist als der vordere (besteht das Wort aus einer geraden Anzahl von Zeichen, wird ein Nullbyte hinzugenommen).

Zum Verschlüsseln eines Bytes wird über zwei Pointer je ein Byte aus den beiden Wortteilen herausgepickt und mit dem Originalbyte EXOR verknüpft. Gegenüber der direkten, einfachen Verwendung des Wortes erhält man so ein wesentlich längeres Verschlüsselungswort, das empfindlicher auf kleine Fehler reagiert. Dadurch wird das Knacken der Verschlüsselung durch versuchsweises Einsetzen aller möglichen Wörter zu einer aussichtslosen Aufgabe. Hier dürfte auch klar werden, daß die Sicherheit der Verschlüsselung mit der Länge des gewählten Wortes steigt.

Wenn immer von Verschlüsselungswort die Rede ist, so sollte man das nicht zu genau nehmen. Besser noch als ein einzelnes Wort ist selbstverständlich ein langer, einprägsamer Satz. Immerhin können bis zu 160 Zeichen eingegeben werden.

Aber selbst falls jemand, wie auch immer, das Wort bzw. den Satz herausbekommt, nützt ihm dies nichts, solange er nicht auch die Verschlüsselungszahl kennt. Diese neunstellige Zahl bildet den Startwert für einen Schieberegister-Zufallszahlengenerator, der $2^{31}-1$ verschiedene Bit-Zustände annehmen kann, also etwa 2^{28} Bytes produziert, ehe sich diese Pseudozufallszahlenfolge wiederholt.

Eine angenehme Eigenschaft der EXOR-Verknüpfung ist, daß sie selbstinvers ist, d.h. ihre nochmalige Benutzung mit denselben Parametern hebt ihre erste Anwendung wieder auf. Daher müssen im Programm Ver- und Entschlüsselung nicht unterschieden werden, abgesehen von der Gewinnung des neuen Dateinamens.

Komprimieren/Expandieren

Menüpunkt 0 schaltet in den Modus Komprimieren/Expandieren um. In dieser Betriebsart wird es möglich sein, Dateien zu verdichten, insbesondere Texte ohne Informationsverlust zu verkürzen. Im Moment macht das Umschalten noch keinen Sinn, denn der entsprechende Programmteil ist noch nicht vorhanden. Er wird in der nächsten Ausgabe des **ATARI *magazins*** abgedruckt. Näheres darüber also im nächsten Heft!

Text- und Zahleneingabe

Hierzu verwendet das Programm übrigens eine recht leistungsfähige Routine, die sich wesentlich angenehmer benutzen läßt als die GEM-Funktion zur String-Eingabe. Die Maximallänge der Eingabe ist nur durch die Größe des Bildschirms begrenzt und wird durch Unterstreichungszeichen angezeigt.

Mittels der Tasten ESC und TAB läßt sich der Cursor schnell an den Anfang bzw. an das Ende des Strings positionieren; bei entsprechend langer Eingabe ist auch die Verwendung aller Pfeiltasten möglich. CLR HOME löscht die bisherige Eingabe, INSERT fügt ein Blank an der Cursor-Position ein. Man kann einen String vorgeben, der zu Beginn erscheinen soll. Nach Setzen bestimmter Flags ist nur noch die Eingabe einer Zahl erlaubt, mit oder ohne Vorzeichen, mit oder ohne Dezimalpunkt.

Michael Schramm

Assemblerlisting

```
*   -----  Assemblerlisting für den ATARI ST  -----

*       Inhalt von Dateien komprimieren
*       und verschlüsseln

*               Autor: Michael Schramm
*               ATARI-Magazin 1987

LF  = 10
CR  = 13
ESC = 27

WORDLEN = 160

        .text

_main   movea.l 4(sp),a5        * A5 auf Basepage.
        lea stackend+2,sp       * Stackp. ans Stackende.
        movea.l 12(a5),a0       * Länge des Textsegments
        adda.l 20(a5),a0        * plus Data-Segmentlänge
        adda.l 28(a5),a0        * plus BSS-Segmentlänge
        adda.w #256,a0          * plus Basepage-Länge
        move.l a0,-(sp)         * auf den Stack, ebenso
        move.l a5,-(sp)         * die Adr. der Basepage.
        clr.w -(sp)             * Dummy.
        move.w #$4a,-(sp)       * Funktionsnummer SETBLCK.
        trap #1                 * Def. Sp.platz reservier.
        adda.w #12,sp           * Stack "aufräumen".
        tst.l d0                * Bei D0 <> 0 ist ein
        beq mem_ok              * Fehler aufgetreten.

error   lea errortxt(pc),a0     * Fehlermeldung
        bsr writeln             * ausgeben.
        move.w #1,-(sp)         * CONIN-Funktionsnummer,
        trap #1                 * auf Tastendruck warten.
ende    clr.w -(sp)             * TERM-Funktionsnummer,
        trap #1                 * Programm beenden.

mem_ok  moveq #$66,d0           * Cursor aus.
        bsr con_esc
        pea dtaddr              * 44 Bytes langer Buffer.
        move.w #$1a,-(sp)       * SETDTA-Funktionsnummer.
        trap #1                 * Funktion aufrufen.
        moveq #-1,d0            * Die Größe des noch ver-
        move.l d0,-(sp)         * fügbaren Speicherplatzes
        move.w #$48,-(sp)       * mit MALLOC feststellen.
        trap #1                 * Ergebnis in D0.
        move.l d0,buff_len      * Buffer-Länge speichern.
        move.l d0,-(sp)         * Länge und Funktionsnr.
        move.w #$48,-(sp)       * auf Stack, Speicherplatz
        trap #1                 * bereitstellen (nur bei
        adda.w #18,sp           * -1 als Länge erfolgt
        tst.l d0                * reine Sp.platz-Abfrage).
        bmi error               * Bei D0 negativ, Fehler.
        movea.l d0,a6           * Sonst enthält D0 Anf.adr
*                               * des Sp.pl., diese -> A6.
        clr.l d7                * D7 als Flag-Register:
*  2: ges. bei Entschlüsseln/Expandieren
*  7: CRP oder CMP
* 15: gültiger Dateiname in fl_name gegeben
* 31: Verschlüsselungsparameter vorhanden
        lea fl_defl(pc),a0      * Defaultwert für
        lea fl_mask1,a1         * Dateimaske 1
copystr1 move.b (a0)+,(a1)+     * in fl_mask1
        bne copystr1            * kopieren.
        lea fl_mask2,a1
copystr2 move.b (a0)+,(a1)+     * Analog
        bne copystr2            * Dateimaske 2.
        clr.b word_mem          * Verschlüsselungsparameter
        clr.b numb_mem          * zunächst Leerstrings.

mainloop lea maintext(pc),a0    * Hier beginnt der Haupt-
        bsr write               * teil des Programms.
        lea fl_mask1,a1
        tst.b d7                * In Abhängigkeit von
        bpl m_textok            * D7-Bit 7 wird der
        lea cmp_text(pc),a0     * richtige Text auf
        lea fl_mask2,a1         * den Bildschirm
m_textok bsr write              * geschrieben.
        lea mask_pos(pc),a0
        bsr write
        movea.l a1,a0
        bsr write
        tst.w d7                * Sofern bereits
        bpl m_wtkey             * vorhanden, wird
        lea name_pos(pc),a0     * der eingegebene
        bsr write               * Dateiname
        lea fl_name,a0          * angegeben.
        bsr write
        bra m_wtkey

m_beep  bsr beep                * Beep bei Falscheingabe.
m_wtkey bsr dirconin            * Auf Tastendruck warten.
        cmpi.b #3,d0            * Bei ^C Progr. beenden.
        beq ende
```

```
         subi.b #'0',d0           * Ansonsten sind nur
         bcs m_beep               * Ziffern zulässig.
         bne m_no_0               * Bei 0 Umschaltung
         bchg #7,d7               * Ver-/Entschlüsseln  -
mainlp1  bra mainloop             * Komprim./Expandieren.

m_no_0   cmpi.b #5,d0             * Nur Eingaben 0 bis 5
         bhi m_beep               * erlaubt, 5 aber nur
         bne m_no_5               * bei Ver-/Entschlüsseln.
         tst.b d7
         bmi m_beep
         lea wordtext(pc),a0      * Bei 5 Eingabe von
         bsr write                * Verschlüsselungswort
         bra word_inp             * und -zahl:

w_beep   bsr beep                 * Beep bei Falscheingabe.
word_inp move.w #WORDLEN,d1       * Eing.-Max.länge in D1,
         move.w #$0300,d0         * Screenposition in D0,
         lea word_mem,a5          * Eingabeadresse in A5.
         bsr instring             * Eingaberoutine.
         cmpi.w #7,d0             * Eingeg. String soll
         bcs w_beep               * mind. Länge 7 haben.
         lsr.w #1,d0              * Halbe Länge in
         move.w d0,-14(a5)        * wordlenl speichern.
         bsr write                * A0 zeigt nun auf
         bra numb_inp             * numbtext!
n_beep   bsr beep                 * Beep bei Falscheingabe.
numb_inp moveq #9,d1              * Wie oben Parameter
         move.w #$0624,d0         * für instring.
         lea numb_mem,a5          * Bit 7 in in_flags
         move.b #128,in_flags     * setzen als Zeichen für
         bsr instring             * numerische Eingabe.
         clr.l d0
numb_lp  move.b (a5)+,d1          * Die zum eingegebenen
         beq numb_end             * Ziffernstring
         lsl.l #1,d0              * gehörende 32-Bit-
         move.l d0,d2             * Binärdarstellung
         lsl.l #2,d0              * in D0 aufbauen.
         add.l d2,d0
         subi.b #'0',d1
         add.l d1,d0
         bra numb_lp

numb_end move.l d0,crp_numl       * Die Zahl abspeichern.
         beq n_beep               * 0 ist unzulässig.
         bset #31,d7              * Zeigt: Verschlüsse-
mainlp2  bra mainlp1              * lungsparam. vorhanden.

m_no_5   subq.b #1,d0             * Menüpunkt 1 erlaubt
         bne m_no_1               * die Eingabe einer
         move.w #$0610,d2         * neuen Dateimaske,
         lea fl_mask1,a5          * je nach Modus in
         tst.b d7                 * fl_mask1 oder fl_mask2.
         bpl name_inp             * Screenposition
         lea fl_mask2,a5          * zunächst in D2.
name_inp moveq #30,d1
         move.w d2,d0             * Eingabe der Dateimaske
         bsr instring             * oder des Dateinamens.
         subq.w #3,d0             * Sollte wenigstens drei
         bcc mainlp2              * Zeichen lang sein.
         bsr beep
         bra name_inp

m_no_1   subq.b #1,d0
         bne m_no_2
         move.w #$0810,d2         * Eingabe des Datei-
         lea fl_name,a5           * namens vorbereiten.
         bset #15,d7              * Zeigt: Dateiname
         bra name_inp             * wurde eingegeben.

m_no_2   tst.w d7                 * Bei Wahl 3 oder 4 muß
m_beep1  bpl m_beep               * Dateiname vorh. sein.
         tst.b d7                 * Zum Ver- oder Ent-
         bmi decstart             * schlüsseln sind
         tst.l d7                 * außerdem die V/E-
         bpl m_beep1              * Parameter vonnöten.
decstart moveq #2,d2              * Bei Verschl./Komprim.
         bclr d2,d7               * D7-Bit-2 rücksetzen,
         subq.b #1,d0             * Bei Entschl./Expand.
         beq cipher1              * setzen.
         bset d2,d7
cipher1  moveq #19,d0             * Auf Zeile 19, Spalte 0
         clr.w d2                 * positionieren.
         bsr conout_ps
         move.w #1,-(sp)          * Parameter für SFIRST.
         lea fl_name,a4
         move.l a4,-(sp)
         move.w #$4e,-(sp)        * SFIRST aufrufen, sucht
         trap #1                  * nach passenden File.
         addq.l #8,sp
         tst.w d0                 * D0 = 0, falls die Suche
         beq fl_found             * erfolgreich war.
         movea.l a4,a0            * Sonst den Dateinamen
         bsr write                * mit einer entsprechenden
         lea ntfndtxt(pc),a0      * Meldung drucken.
         bsr writeln
fl_ready lea waittext(pc),a0      * Zum Tastendruck
         bsr write                * auffordern und
         bsr dirconin             * entsprechend warten.
         bra mainloop             * Weiter im Hauptteil.

fl_found lea dtaddr,a3            * Die 44 Bytes ab dtaddr
         lea 30(a3),a0            * enthalten jetzt wesent-
         moveq #14,d3             * liche Informationen.
fl_pr_lp move.b (a0)+,d0          * Zunächst den Namen
         bne flpr_chr             * drucken mit nachfol-
         subq.l #1,a0             * genden Blanks,
         moveq #32,d0             * insgesamt 15 Stellen.
flpr_chr bsr writechr
         dbf d3,fl_pr_lp
         move.l 26(a3),d0         * Länge des Files. Bei 0
         beq nxt_file             * nächstes File vornehmen.
         move.l d0,d6             * Länge in D6 merken.
         add.l #514,d0            * Das File muß komplett
         cmp.l buff_len,d0        * in den Buffer passen,
         lea flsrtext(pc),a0      * so daß wenigstens noch
         bcc errbreak             * 512 Bytes als Schreib-
*                                 * Buffer übrig bleiben.
         movea.l a6,a0            * Buffer-Adr. -> A0.
         movea.l a4,a1            * A1 auf fl_name.
         movea.l a0,a2            * Im eingegebenen Datei-
dir_loop move.b (a1)+,d0          * namensuchausdruck wird
         move.b d0,(a0)+          * die Adresse des letzten
         beq dir_end              * Backslash ermittelt.
         cmpi.b #'\',d0           * Die Position+1 wird in
         bne dir_loop             * A2 festgehalten und
         movea.l a0,a2            * hinter diesen Character
         bra dir_loop             * der von SFIRST bzw.
dir_end  lea 30(a3),a1            * SNEXT gelieferte File-
dirfl_lp move.b (a1)+,(a2)+       * name gehängt, denn der
         bne dirfl_lp             * ist Directory-lokal,
*                                 * also ohne Pfad.
         clr.w -(sp)              * Nur zum Lesen öffnen.
         move.l a6,-(sp)          * Adresse des Filenamens.
         move.w #$3d,-(sp)        * Funktionsnummer OPEN.
         trap #1                  * Sofern Öffnen möglich
         addq.l #8,sp             * war, bekommt man in D0
         tst.w d0                 * eine "handle-Nummer"
         bpl open_ok              * zurückgeliefert.
         lea operrtxt(pc),a0      * Bei D0 negativ,
errbreak lea stackend+2,sp        * liegt ein Fehler vor.
         bsr write
         bra nxt_file

open_ok  move.w d0,-(sp)          * handle-Nr. für CLOSE.
         move.l a6,-(sp)          * Bufferadresse.
         move.l d6,-(sp)          * Anz. zu lesender Bytes.
         move.w d0,-(sp)          * handle-Nr. des Files.
         move.w #$3f,-(sp)        * READ-Funktionsnummer.
         trap #1                  * In D0 wird die Anzahl
         addq.l #4,sp             * der tatsächlich gelese-
         cmp.l (sp)+,d0           * nen Bytes zurückgelie-
         addq.l #4,sp             * fert. Falls diese <>
         beq fl_close             * gewünschter Zahl, ist
         lea rderrtxt(pc),a0      * ein Fehler eingetreten.
         bsr write                * Fehlermeldung ausgeben.
fl_close move.w #$3e,-(sp)        * CLOSE-Funktionsnummer.
         trap #1                  * Close gibt die handle-
         addq.l #4,sp             * Nr. etc. wieder frei.
         lea wordpntl,a0
         clr.l (a0)+              * Codierparameter
         move.l 4(a0),(a0)        * initialisieren.
         lea workarea,a0
         clr.w -(sp)              * Param. für CREATE.
         move.l a0,-(sp)          * Bei Verschl./Komprim.
         btst #2,d7               * Namen der zu kreieren-
         beq cipher2              * den Datei bei cipher2
         movea.l a6,a1            * bestimmen. Bei Expand.
         tst.b d7                 * muß 1. Byte 1-3 sein,
         bpl deciph1              * der Name steht in
         addq.l #4,a1             * Klartext ab dem 4.
         move.b (a6),d0           * Byte in der Datei.
         beq unsuited             * Falls das 1. Byte
         subq.b #4,d0             * nicht den Anforderungen
         bcs deciph1              * genügt, darf Expansion
unsuited lea unstdtxt(pc),a0      * nicht stattfinden.
         bra errbreak

deciph1  moveq #19,d1             * Max. 20 Bytes lesen, um
copnm_lp move.b (a1)+,d0          * den Namen zu bestimmen.
         tst.b d7                 * Bei Entschlüsselung
         bmi coprnexp             * muß auch der Name
         bsr code_d0              * entschlüsselt werden.
coprnexp move.b d0,(a0)+          * Name in workarea.
         dbeq d1,copnm_lp         * Falls kein Nullbyte
         bne unsuited             * zu finden, ungeeignet.
         move.l a1,a5             * Adr. Dateiinhalt.
         move.l (sp),a0           * #workarea
         moveq #3,d0              * Nun prüfen, ob der
tname_lp move.b (a0)+,d2          * String ein Dateiname
         beq flcrendl             * sein kann.
         cmpi.b #'.',d2           * Ein Dateiname darf
         bne trenopnt             * höchstens einen
         tas d1                   * Punkt enthalten.
         bmi unsuited
         bra tname_lp
trenopnt tst.b d1                 * Die Extension darf
         bpl trebfpnt             * nicht länger als 3
         subq.w #1,d0             * Zeichen lang sein.
         bcs unsuited
trebfpnt cmpi.b #'_',d2           * Außer dem Punkt
         beq tname_lp             * sind nur Buchstaben,
         cmpi.b #'0',d2           * Ziffern und Unter-
         bcs unsuited             * streichungszeichen
         cmpi.b #'z'+1,d2         * zugelassen.
         bcs tname_lp
         bra unsuited

cipher2  lea fl_mask1,a1          * Bei Verschl./Komprim.
         tst.b d7                 * wird der neue Dateiname
         bpl fl_cr_lp             * aus der Dateimaske
         lea fl_mask2,a1          * gewonnen.
fl_cr_lp move.b (a1)+,d0
         beq flcr_end
         cmpi.b #'*',d0           * Ein Sternchen in der
         beq flcr_ast             * Maske wird durch
         move.b d0,(a0)+          * den Namen der gelesenen
         bra fl_cr_lp             * Datei ohne Extension
flcr_ast lea 30(a3),a5            * ersetzt.
flcrname move.b (a5)+,d0
         beq fl_cr_lp
         cmpi.b #'.',d0
         beq fl_cr_lp
         move.b d0,(a0)+
         bra flcrname

flcr_end clr.b (a0)+              * Nullbyte beendet Namen.
flcrendl lea arrowtxt(pc),a0      * ' — ' ausgeben.
         bsr write
         movea.l (sp),a0          * Dateinamen drucken.
         bsr write
         move.w #$3c,-(sp)        * CREATE ausführen.
         trap #1
         addq.l #8,sp
         tst.w d0                 * Bei D0 >= 0 ist alles
         bpl createok             * in Ordnung.
         lea crerrtxt(pc),a0      * Sonst Fehlermeldung.
         bra errbreak

createok lea handle,a0            * handle-Nr. merken und
         move.w d0,(a0)           * den Write-Buffer-
         clr.l -(a0)              * Pointer nullsetzen.
         move.l d6,d0             * Dateilänge in D0, auf-
         addq.l #1,d0             * runden auf nächste
         bclr #0,d0               * gerade Zahl.
         move.l 6(a0),d1          * Ges.bufferlänge in D1.
         sub.l d0,d1              * Davon ben. Länge abz.
         andi.w #-512,d1          * Abrunden auf 512-Vielf.
         move.l d1,-(a0)          * Ergebnis in cr_b_len.
         add.l a6,d0              * Bufferadr. für neues
         move.l d0,-(a0)          * File in cr_b_adr.
         tst.b d7                 * Je nach gewähltem
         bmi compress             * Modus weiter.

* Nun Ver- bzw. Entschlüsseln des Dateiinhalts:

         movea.l a5,a0            * Bei Entschlüsselung
         btst #2,d7               * wird der Dateiname
         bne flcontcd             * übersprungen.
         lea 30(a3),a0            * A0 auf den Dateinamen
cd_nm_lp move.b (a0),d0           * in dtaddr.
         bsr put_cdbt             * Den Namen Byte für Byte
         tst.b (a0)+              * verschl. und in den
         bne cd_nm_lp             * Write-Buffer schreiben.
         movea.l a6,a0            * A0 auf Bufferbeginn.
flcontcd move.l a0,d0             * Den Dateilängenzähler
         sub.l a6,d0              * D6 um die Zahl der ggf.
         sub.l d0,d6              * Überspr. Bytes vermind.
cd_fl_lp move.b (a0)+,d0
         bsr put_cdbt             * Den Dateiinhalt
         subq.l #1,d6             * byteweise
         bhi cd_fl_lp             * verschlüsseln.

cr_close bsr wr_buff              * Bufferrestinhalt
crclose1 move.w handle,-(sp)      * abspeichern.
         move.w #$3e,-(sp)        * CLOSE
         trap #1                  * neue Datei.
         addq.l #4,sp

nxt_file bsr newline              * Zeile abschließen.
         move.w #$4f,-(sp)        * SNEXT, sucht nächstes
         trap #1                  * File, dessen Name zum
         addq.l #2,sp             * eingegeb. Muster paßt.
         tst.l d0                 * Falls D0 <> 0, keine
         beq fl_found             * weiteren Files
         bra fl_ready             * vorhanden.

compress bra crclose1             * Hier wird die
* Komprimierungs/Expandierungsroutine ergänzt werden.

code_d0  movem.l d2-d4/a0,-(sp)   * D0.B verschlüsseln:
         lea wordlenl,a0
         move.w (a0)+,d2          * wordlenl in D2.
         move.w (a0),d3           * wordpntl in D3.
         addq.w #1,d3             * wordpntl inkrement.
         cmp.w d2,d3              * wordpntl modulo
         bcs cd_p1_ok             * wordlenl nehmen,
         clr.w d3                 * also ggf. nullsetzen.
cd_p1_ok move.w d3,(a0)+          * wordpntl abspeichern.
         move.b 10(a0,d3.w),d4    * D0 mit Byte aus der
         eor.b d4,d0              * 1. Worthälfte verschl.
         move.w (a0),d3           * wordpnt2 in D3.
         addq.w #1,d3             * wordpnt2 inkrement.
         cmp.w d2,d3              * wordpnt2 modulo
         bls cd_p2_ok             * wordlenl+1 nehmen.
         clr.w d3
cd_p2_ok move.w d3,(a0)+          * wordpnt2 abspeichern.
         add.w d2,d3              * D0 nun mit einem Byte
         move.b 8(a0,d3.w),d4     * aus der 2. Worthälfte
         eor.b d4,d0              * verschlüsseln.
         move.l (a0),d2           * crp_numb in d2.
         moveq #7,d4              * Acht Durchläufe:
cd_rotat move.w d2,d3             * D2 wird als 31-Bit-
         lsr.w #3,d3              * Schieberegister
         eor.w d2,d3              * benutzt. Das Register
         lsl.l #1,d2              * wird rechtsgeschoben.
         roxr.w #1,d3             * Als neues Bit wird
         roxr.l #2,d2             * das Ergebnis der EXOR-
         dbf d4,cd_rotat          * Verknüpfung der Bits 0
         move.l d2,(a0)           * & 3 vor dem Schieben
         eor.b d2,d0              * eingeschoben. D0.B
         movem.l (sp)+,d2-d4/a0   * wird mit zur Verschl.
         rts                      * herangezogen.

ptb_full bsr wr_buff              * Wenn Writebuffer voll,
         bra putbyte1             * zunächst abspeichern.
put_cdbt bsr code_d0              * D0.B verschlüsseln.
put_byte movem.l d1-d2/a0-a1,-(sp)
putbyte1 lea cr_b_adr,a0          * D0.B wegschreiben:
         movea.l (a0)+,a1         * cr_b_adr in A1.
         move.l (a0)+,d1          * cr_b_len in D1.
         move.l (a0),d2           * cr_b_pnt in D2.
         cmp.l d1,d2              * Prüfen, ob der
         beq ptb_full             * Buffer voll ist.
         move.b d0,0(a1,d2.l)     * Das Byte in den
         addq.l #1,d2             * Buffer schreiben,
         move.l d2,(a0)           * den Pointer inkrem.
         movem.l (sp)+,d1-d2/a0-a1
         rts

wr_buff  move.w d0,-(sp)          * Writebuffer schreiben:
         lea cr_b_adr,a0
         move.l (a0),-(sp)        * cr_b_adr auf Stack.
         addq.l #8,a0
         move.l (a0),-(sp)        * cr_b_pnt = akt. Länge.
         clr.l (a0)+              * cr_b_pnt auf 0.
         move.w (a0),-(sp)        * handle-Nr. auf Stack.
         move.w #$40,-(sp)        * WRITE-Funktionsnummer.
         trap #1                  * Schreiben des Buffer-
         addq.l #4,sp             * inhalts. Als Ergebnis
         cmp.l (sp)+,d0           * erhält man die Anzahl
         addq.l #4,sp             * der tatsächlich
         bne wr_error             * geschriebenen Bytes.

         move.w (sp)+,d0          * Weicht diese von der
         rts                      * geforderten Zahl ab,
*                                 * liegt ein Fehler vor.

wr_error lea stackend+2,sp        * Stack aufräumen,
         lea wrerrtxt(pc),a0      * Fehlermeldung
         bsr write                * ausgeben und das
         bra crclose1             * File schließen.

* ———————— Allgemeine I/O-Routinen ————————

writeln  bsr write                * String ausgeben.
newline  moveq #CR,d0             * CR ausgeben.
```

```
         bsr writechr
         moveq #LF,d0             * LF ausgeben.
writechr movem.l d0-d4/a0-a4,-(sp)
         move.w d0,-(sp)          * Char. auf Stack.
         move.w #2,-(sp)          * Parameter für Ausgabe
         trap #1                  * auf den Bildschirm.
         addq.l #4,sp             * Stack aufräumen.
         movem.l (sp)+,d0-d4/a0-a4
return   rts

write    move.b (a0)+,d0          * Einen String drucken,
         beq return               * der bei A0 beginnt und
         bsr writechr             * durch ein Nullbyte
         bra write                * begrenzt ist.

* Eingabe eines Strings der max. Länge D1.W. D0-Lowbyte
* gibt die X-Pos. an, D0-Lowwordhighbyte die Y-Pos. Der
* String wird in den Speicher ab A5 und auf den Bildschirm
* geschrieben. Als Ergebn. Stringlänge in D0.W. Der String
* ist durch ein Nullbyte terminiert. Der alte Speicherinh.
* wird als Standardeingabe vorgegeben. in_flags beachten!

instring movem.l r2-r14,-(sp)
         move.w d0,d3
         moveq #$76,d0            * Automatischer
         bsr con_esc              * Überlauf ein.
         clr.w d4
         move.b d3,d4             * D3.w = ypos.
         lsr.w #8,d3              * D4.W = xpos.
         lea in_flags(pc),a3      * A3 = #in_flags
in_start move.b d3,d0
         move.b d4,d2             * Auf Stringanfang
         bsr conot_ps             * positionieren.
         clr.w d5                 * Länge des vorgegebenen
         move.w d1,d2             * Strings feststellen
         beq in_tlend             * (in D5.W), ihn dabei
in_tslen addq.w #1,d5             * drucken und den Rest
         move.b -1(a5,d5.w),d0    * des Eingabefeldes mit
         bne in_tlmre             * Unterstreichungszeichen
         subq.w #1,d5             * auffüllen.
         moveq #$5f,d0 * '_'
in_tlmre bsr conout
         subq.w #1,d2
         bgt in_tslen
in_tlend movea.w d5,a6
in_mloop moveq #80,d2
         cmp.w a6,d5        * Ab hier fogende Reg.belegung:
         bcc in_nogrw       * D1 = maximal zulässige Länge
         move.w a6,d5       * D3/D4 = YPOS/XPOS Stringanfang
in_nogrw move.l a6,d0       * D5 = aktuelle Stringlänge
         add.w d4,d0        * A3 = #in_flags
         add.w d2,d0        * A5 = Adresse des Eingabefeldes
         divu d2,d0         * A6 = akt. Pos. im Str. (0..D5)
         subq.w #1,d0
         add.w d3,d0
         move.l d0,d2
         swap d2            * D2.W = akt. X-Position
         move.l d0,d6       * D6.W = akt. Y-Position
         bsr conot_ps
         bclr #5,(a3)             * Nur sofern der String
         beq in_wtchr             * nach dem letzten Aus-
         lea 0(a5,a6.w),a0        * drucken verändert
         move.w d5,d7             * worden ist, signali-
         sub.w a6,d7              * siert durch Bit 5 von
         beq in_empty             * in_flags, wird der
in_ploop move.b (a0)+,d0          * Stringteil ab der
         bsr conout               * aktuellen Positon
         subq.w #1,d7             * ausgedruckt.
         bgt in_ploop
in_empty move.w d1,d7
         sub.w d5,d7
         beq in_mxler
         moveq #$5f,d0 * '_'
         bsr conout
in_mxler move.w d6,d0
         bsr conot_ps
in_wtchr moveq #$65,d0            * Cursor ein.
         bsr con_esc
         bsr dirconin
         move.l d0,d7
         moveq #$66,d0            * Cursor aus.
         bsr con_esc
         move.l d7,d0
         cmpi.b #32,d0            * Control-Character
         bcs in_crtl              * gesondert behandeln.
         cmpi.b #$7f,d0           * Ebenso DELETE.
         beq in_del               * Alles andere gilt
         tst.b (a3)               * als druckbar.
         bpl in_nonum
         cmpi.b #$2d,d0 * '-'
         bne in_nomin             * Falls nur numerische
         btst #4,(a3)             * Eingabe gewünscht,
         bne in_nonum             * wird hier auf Zuläs-
in_errjp bra in_error             * sigkeit geprüft.
in_nomin cmpi.b #$2e,d0 * '.'
         bne in_nopnt
         btst #3,(a3)
         bne in_nonum
         bra in_errjp
in_nopnt cmpi.b #$30,d0
         bcs in_errjp
         cmpi.b #$3a,d0
         bcc in_errjp       * Sobald das Eingabefeld voll
in_nonum cmp.w a6,d1        * ist, werden keine normalen
         beq in_errjp       * Eingaben mehr akzeptiert.
         move.b d0,0(a5,a6.w) * Zeichen in Eingabefeld.
         bsr conout         * Das Zeichen drucken.
         addq.w #1,a6       * Auf nächste Position.
in_mlp1  bra in_mloop
in_del   move.w d5,d0             * Das Zeichen unter dem
         sub.w a6,d0              * Cusor ausfügen. Das ist
         beq in_mlp1              * nur möglich, falls der
         lea 0(a5,a6.w),a0        * Cursor nicht hinter der
         lea 1(a0),a1             * bisherigen Eingabe
in_dellp move.b (a1)+,(a0)+  * steht.
         dbf d0,in_dellp
         subq.w #1,d5
in_bset5 bset #5,(a3)
         bra in_mlp1

in_crtl  move.b d0,d7
         beq in_zero
         subq.b #8,d7
         bne in_nobs
in_bs    subq.w #1,a6             * Bei BS die Cursorposition
         move.w a6,d0             * um 1 verringern und DELETE
         bpl in_del               * ausführen. Das ist nur
         addq.w #1,a6             * möglich, sofern der Cursor
         bra in_mlp1              * nicht am Anfang steht.

in_nobs  subq.b #1,d7
         bne in_notab             * Bei TAB Cursor ans
         move.w d5,a6             * Eingabeende.
in_mlp2  bra in_mlp1

in_notab cmpi.b #ESC,d0
         bne in_noesc             * Bei ESC Cursor an
         clr.w a6                 * den Eingabeanfang.
         bra in_mlp2

in_noesc subq.b #4,d7         * Von den übrigen Steuer-
         bne in_error         * codes ist nur CR erlaubt.
         clr.b 0(a5,d5.w)     * Nullbyte als Str.begrenzung.
         move.b (a3),d7       * instring-Flags in D7.
         bpl in_term
         movea.l a5,a0
in_ntst  move.b (a0)+,d0
         beq in_nuend             * Wenn eine rein numerische
         cmpi.b #'-',d0           * Eingabe gewünscht war,
         beq in_tmin              * wird der String nun
         cmpi.b #'.',d0           * darauf untersucht, ob er
         beq in_tpnt              * eine Zahl mit den
         cmpi.b #'0',d0           * gewünschten Eigenschaften
         bcs in_error             * darstellt.
         cmpi.b #'9'+1,d0
         bcc in_error             * D7-Bit-7 rücksetzen,
         bclr #7,d7               * sobald eine Ziffer
         bra in_ntst              * gefunden ist.

in_tpnt  bclr #3,d7               * Nur ein Dezimalpunkt
         bne in_ntst              * pro Zahl ist erlaubt.
         bra in_error

in_tbeg  movea.l a0,a1            * Ein nicht am Ende des
         subq.l #1,a1             * Strings gefundenes
         cmpa.l a1,a5             * Minuszeichen muß am
         beq in_ntst              * Anfang stehen.
in_error bsr beep
in_mlp3  bra in_mlp2

in_tmin  bclr #4,d7               * Nur ein Minuszeichen
         beq in_error             * pro Zahl ist erlaubt.
         tst.b (a0)               * Dieses muß am Anfang
         bne in_tbeg              * oder am Ende der Zahl
in_nuend tst.b d7                 * stehen.
         bmi in_error             * Der Str. muß wenigstens
in_term  move.w d3,d0             * eine Ziffer enthalten.
         move.w d4,d2
         bsr conot_ps
         movea.l a5,a0            * Der String wird ab
         bsr write                * der gewählten
         move.w d1,d2             * Screen-Position
         sub.w d5,d2              * gedruckt. Falls er
         beq in_fllen             * nicht die volle
in_flclr moveq #32,d0             * Länge aufweist,
         bsr conout               * wird das Restfeld
         subq.w #1,d2             * mit Blanks gefüllt.
         bhi in_flclr
in_fllen move.w d5,d0
         clr.b (a3)
         movem.l (sp)+,r2-r14
         rts

in_zero  swap d0                  * Oberes Word untersuchen.
         subi.b #$47,d0           * Bei CLR HOME einen
         bne in_noclr             * Leerstring erzeugen
         clr.b (a5)               * und zum UP-Anfang
         bra in_start             * verzweigen.

in_noclr subq.b #1,d0
         bne in_noup              * Bei "Pfeil aufwärts"
         moveq #80,d0             * 80 Zeichen zurück,
in_minus cmpa.w d0,a6             * falls möglich.
         bcs in_mlp4
in_minsl suba.w d0,a6
in_mlp4  bra in_mlp3

in_noup  subq.b #3,d0
         bne in_nolft             * Bei "Pfeil links"
         moveq #1,d0              * ein Zeichen zurück,
         bra in_minus             * falls möglich.

in_nolft subq.b #2,d0
         bne in_norgt             * Bei "Pfeil rechts"
         moveq #1,d0              * ein Zeichen vor,
in_adda6 adda.w d0,a6             * falls möglich.
         cmp.w a6,d5
         bcc in_mlp4
         bra in_minsl

in_norgt subq.b #3,d0
         bne in_nodwn             * Bei "Pfeil abwärts"
         moveq #80,d0             * 80 Zeichen vor,
         bra in_adda6             * falls möglich.

in_nodwn subq.b #2,d0             * Außer INSERT ist kein
         bne in_error             * weiterer Code erlaubt.
         cmp.w d1,d5              * Nicht möglich, falls
         beq in_mlp4              * Eingabe Höchstlänge hat.
         lea 1(a5,d5.w),a0
         lea -1(a0),a1
         move.w d5,d0             * Den Stringteil hinter dem
         sub.w a6,d0              * Cursor um eine Stelle
         beq in_inend             * nach rechts schieben,
in_inslp move.b -(a1),-(a0)       * sofern der Cursor nicht
         subq.w #1,d0             * hinter der bisherigen
         bne in_inslp             * Eingabe steht.
in_inend move.b #32,(a1)          * Ein Blank einfügen.
         addq.w #1,d5             * Stringlänge erhöhen.
         bra in_bset5

dirconin movem.l d1-d4/a0-a4,-(sp)
         move.w #7,-(sp)              * Zeichen von der
         trap #1                      * Tastatur lesen,
         addq.l #2,sp                 * ohne Echo.
dircirts movem.l (sp)+,d1-d4/a0-a4
         rts

beep     moveq #7,d0                  * BEEP-Signal bei
         bra conout                   * Falscheingaben.

con_esc  move.l d0,-(sp)              * ESC ausgeben,
         moveq #ESC,d0                * dann D0.
         bsr conout
         move.l (sp)+,d0

conout   movem.l d0-d4/a0-a4,-(sp)
         move.w d0,-(sp)              * D0 als Character
         move.w #2,-(sp)              * über GEM-TRAP
         trap #1                      * ausgeben.
         addq.l #4,sp
         movem.l (sp)+,d0-d4/a0-a4
         rts

conot_ps move.l d0,-(sp)              * PRINT AT d0,d2:
         moveq #$59,d0
         bsr con_esc                  * Es wird ESC,'Y',
         move.l (sp)+,d0              * D0.B+32,D2.B+32
         bsr conot_32                 * ausgegeben.
         move.b d2,d0
conot_32 addi.b #32,d0
         bra conout


* ------------------ Datenbereich ----------------------

in_flags  .dc.b 0                     * Flags für instring:
* Die Bedeutung der Bits:
* 3: Punkt zulässig bei numerischer Eingabe
* 4: Minuszeichen zul. bei numer. Eingabe
* 5: zeigt Änderung an, intern benutzt
* 7: numerische Eingabe

maintext  .dc.b ESC,'E',LF,' Dateiverschlüsselungsprogr'
          .dc.b 'amm von Michael Schramm, ATARI-Magazin'
          .dc.b ESC,'Y',38,33,'1: Dateimaske:',CR,LF,LF
          .dc.b ' 2: Dateiname:',ESC,'Y',48,33,'^C: Ende'
          .dc.b ESC,'Y',36,33,'0: -> Datei',0
crp_text  .dc.b 'komprimierung',ESC,'Y',42,33,'3: Versch'
          .dc.b 'lüsselung',CR,LF,LF,' 4: Entschlüsselung'
          .dc.b CR,LF,LF,' 5: Verschlüsselungswort und '
          .dc.b '-zahl eingeben',0
cmp_text  .dc.b 'verschlüsselung',ESC,'Y',42,33,'3: Kom'
          .dc.b 'primieren',CR,LF,LF,' 4: Expandieren',0
mask_pos  .dc.b ESC,'Y',38,48,0
name_pos  .dc.b ESC,'Y',40,48,0
wordtext  .dc.b ESC,'E',LF,' Verschlüsselungswort '
          .dc.b '(mindestens 7 Zeichen):',0
numbtext  .dc.b ESC,'Y',38,33,'Verschlüsselungszahl '
          .dc.b '(ungleich 0):',0

errortxt  .dc.b 'Fehler bei Speicherplatzreservierung'
waittext  .dc.b CR,LF,LF,'        <Return>',0
ntfndtxt  .dc.b '  nicht gefunden',0
flsztext  .dc.b 'zu groß',0
operrtxt  .dc.b 'nicht zu öffnen',0
unstdtxt  .dc.b 'ungeeignet',0
rderrtxt  .dc.b ' - Lesefehler',0
crerrtxt  .dc.b ' - CREATE klappt nicht',0
wrerrtxt  .dc.b ' - Schreibfehler',0
arrowtxt  .dc.b '--> ',0
fl_def1   .dc.b '*.CRP',0
fl_def2   .dc.b '*.CMP',0


* ------------------ BSS-Bereich ----------------------

          .bss
cr_b_adr  .ds.l 1   * Adresse des Writebuffers.
cr_b_len  .ds.l 1   * Länge des Writebuffers.
cr_b_pnt  .ds.l 1   * Pointer im Writebuffer.
handle    .ds.w 1   * handle-Nr. des neuen Files.
buff_len  .ds.l 1   * Länge des gesamten Buffers.
wordlenl  .ds.w 1   * Länge der 1. Hälfte des V/E-Wortes.
wordpnt1  .ds.w 1   * Pointer in 1. Hälfte des Wortes.
wordpnt2  .ds.w 1   * Pointer in 2. Hälfte des Wortes.
crp_numb  .ds.l 1   * Aktuelle Verschlüsselungszahl.
crp_numl  .ds.l 1   * Startwert der Verschl.zahl.
word_mem  .ds.b WORDLEN+2 * Speicher für V/E-Wort.
stack     .ds.w 199  * Platz für den Stack.
stackend  .ds.w 1
dtaddr    .ds.b 44  * File-Attribute-Buffer.
fl_name   .ds.b 60  * Speicher für Dateiname.
fl_mask1  .ds.b 31  * Speicher für Verschl.-Dateimaske.
fl_mask2  .ds.b 31  * Speicher für Komprim.-Dateimaske.
numb_mem  .ds.b 10  * Speicher für V/E-Zahlstring.
workarea  .ds.b 80  * Arbeitsraum.

          .end
```

Beim Listing zum Kurs "1050-Floppy mit Happy-Enhancement" unterlief uns leider ein Fehler. Hier das richtige Listing. Wir bitten um Entschuldigung!

Listing 1

```
        .OPT NO LIST
;
; LISTING 1
;
; PROGRAMMIERT DIE FLOPPY MIT
; DEM NEUEN X-BEFEHL, DER IM
; SPEICHER BEI BEFADR STEHEN MUSS
;
DSBI    =   #0300
DDRV    =   #0301
DKMD    =   #0302
DSTA    =   #0303
DPUF    =   #0304
DTIO    =   #0306
DLEN    =   #0308
DAUX    =   #030A
SIO     =   #E459
;
PUF     =   #0600
BEFTAB  =   #9780
BEFADR  =   #9000
;
        *=  #A000
;
; BEFEHLSTABELLE LADEN
;
        LDX # <PUF
        LDY # >PUF
        JSR SETPUF
        LDX # <BEFTAB
        LDY # >BEFTAB
        JSR SETAUX
        LDA #'R
        JSR DOSIO
;
; LEEREN BEFEHLSPLATZ SUCHEN
;
        LDY ##20
SUCH    LDA PUF,Y
        BEQ LEER
        INY
        JMP SUCH
;
; NEUEN BEFEHL EINFUEGEN
;
LEER    LDA #'X      ; NEUER BEFEHL
        STA PUF,Y    ; X
        LDA # <BEFADR
        STA PUF+#20,Y ; BEFEHLSADR.
        LDA # >BEFADR ; SETZEN
        STA PUF+#40,Y
;
; NEUE TABELLE ZUR FLOPPY
;
        LDA #'P
        JSR DOSIO

;
; NEUEN BEFEHL ZUR FLOPPY
; SCHICKEN
; ( COMPUTER-RAM  #9000-#93FF )
; ( IN FLOPPY-RAM #9000-#93FF )
;
        LDX # <BEFADR
        LDY # >BEFADR
        JSR SETPUF
        JSR SETAUX
PROG    LDA #'P
        JSR DOSIO
        CLC
        LDA DPUF
        ADC ##80
        STA DPUF
        BCC *+5
        INC DPUF+1
        LDA DPUF
        STA DAUX
        LDA DPUF+1
        STA DAUX+1   ; KOMMANDO KANN
        CMP ##94     ; BIS #93FF
        BCC PROG     ; GEHEN
;
; RAMBEREICH IN FLOPPY SCHUETZEN
;
        LDX ##60
        LDY ##60
        JSR SETAUX
        LDA #'H
        JSR DOSIO
;
; FERTIG, FLOPPY PROGRAMMIERT
;
        RTS
;
SETPUF  STX DPUF
        STY DPUF+1
        RTS
SETAUX  STX DAUX
        STY DAUX+1
        RTS
DOSIO   LDX #0
        CMP #'P
        BNE *+4
        LDX ##80
        CMP #'R
        BNE *+4
        LDX ##40
        STX DSTA
        STA DKMD
        LDA #'1
        STA DSBI
        LDA #1
        STA DDRV
        LDA #2
        STA DTIO
        LDA #128
        STA DLEN
        LDA #0
        STA DTIO+1
        STA DLEN+1
        JSR SIO
        BMI FEHLER
        RTS
;
FEHLER  LDA 710      ;BEI FEHLER
        PHA          ;HINTERGRUND-
        SBC #8       ;FARBE AENDERN
        STA 710
        LDA #0
        STA 20
WARTE   LDA 20       ;WARTEN
        CMP #50
        BCC WARTE
        PLA          ;ALTE HINTER-
        STA 710      ;GRUNDFARBE
        PLA
        PLA
        RTS
;
; STARTADRESSE
;
        *=  #02E0
        .WORD #A000
```

Listing 2

```
        .OPT NO LIST
;
; LISTING 2
;
; DIESER BEFEHL ERMOEGLICHT DAS
; DAS AUSLESEN DES FLOPPYBETRIEB-
; SYSTEMS
;
DAUX    =   #82
PUFL    =   #99
PUFH    =   #9A
RAMPUF  =   #8000
;
SENDCPL =   #F48F
SENDPUF =   #F503
;
        *=  #9000   ; KOMMANDOADR.
;
        LDA DAUX+1   ; BEI POSITIVEM
        BMI NOSWITCH ; DAUX+1
        ORA ##80     ; 2. 4K ROM
        STA DAUX+1   ; EINBLENDEN
;
; BLENDE 2. 4K ROM EIN
;
        LDA #FFF9
;
NOSWITCH LDY #0
LOOP    CPY ##F8     ; AUFPASSEN,
        BCC OK       ; DASS NICHT
        CPY ##FA     ; AUS VERSEHEN
        BCS OK       ; UMGEBLENDET
        LDA DAUX+1   ; WIRD
        CMP ##FF     ; ( LDA #FFF8 )
        BNE OK       ; ( LDA #FFF9 )
        JMP WEITER
;
OK      LDA (DAUX),Y
        STA RAMPUF,Y
WEITER  INY
        BNE LOOP
;
; BLENDE 1. 4K ROM EIN
;
        LDA #FFF8
;
; SIGNALISIERE COMPUTER, DASS
; OPERATION BEENDET IST UND
; DATENBLOCK GLEICH FOLGT
;
        JSR SENDCPL
;
; SENDE 256 DATENBYTES MIT
; CHECKSUMME
;
        LDA # <RAMPUF
        STA PUFL
        LDA # >RAMPUF
        STA PUFH
        LDY #0
        JSR SENDPUF
;
; ZURUECK ZUM SYSTEM
;
        RTS
```

Listing 3

```
        .OPT NO LIST
;
; LISTING 3
;
; LAEDT DURCH DEN X-BEFEHL DAS
; FLOPPY BETRIEBSSYSTEM IN DEN
; COMPUTER
; ( 1. 4K NACH PUF1 )
; ( 2. 4K NACH PUF2 )
;
DSBI    =   #0300
DDRV    =   #0301
DKMD    =   #0302
DSTA    =   #0303
DPUF    =   #0304
DTIO    =   #0306
DLEN    =   #0308
DAUX    =   #030A
SIO     =   #E459
;
PUF1    =   #6000
PUF2    =   #7000
;
        *=  #5000
;
        LDA # <PUF1 ; 1. 4K ROM
        STA DPUF    ; NACH PUF1
        LDA # >PUF1
        STA DPUF+1
        LDA # <#F000 ; NEG. DAUX+1
        STA DAUX     ;  SPRICHT
        LDA # >#F000 ; 1. 4K ROM
        STA DAUX+1   ;  AN
;
LOOP1   JSR DOSIO    ; LADE 1. 4K ROM
        INC DPUF+1   ; IN COMPUTER
        INC DAUX+1
        LDA DAUX+1
        BNE LOOP1
;
        LDA # <PUF2 ; 2. 4K ROM
        STA DPUF    ; NACH PUF2
        LDA # >PUF2
        STA DPUF+1
        LDA # <#7000 ; POS. DAUX+1
        STA DAUX     ;  SPRICHT
        LDA # >#7000 ; 2. 4K ROM
        STA DAUX+1   ;  AN
;
LOOP2   JSR DOSIO    ; LADE 2. 4K ROM
        INC DAUX+1   ; IN COMPUTER
        INC DPUF+1
        LDA DAUX+1
        CMP ##80
        BNE LOOP2

;
; DAS FLOPPY-ROM BEFINDET SICH
; JETZT IM COMPUTER SPEICHER
; 1. 4K BEI PUF1
; 2. 4K BEI PUF2
;
        RTS
;
DOSIO   LDA #'1
        STA DSBI
        LDA #1
        STA DDRV
        LDA #'X      ; DAS NEUE
        STA DKMD     ; KOMMANDO
        LDA ##40     ; STATUS READ
        STA DSTA
        LDA #2
        STA DTIO
        LDA #0
        STA DTIO+1   ; 256 BYTES
        STA DLEN     ; WERDEN VON
        LDA #1       ; DER FLOPPY
        STA DLEN+1   ; ERWARTET
        JSR SIO
        BMI FEHLER
        RTS
;
FEHLER  LDA 710      ; BEI FEHLER
        PHA          ; HINTERGRUND-
        SBC #8       ; FARBE AENDERN
        STA 710
        LDA #0
        STA 20
WARTE   LDA 20       ; WARTEN
        CMP #50
        BCC WARTE
        PLA          ; ALTE HINTER-
        STA 710      ; GRUNDFARBE
        PLA
        PLA
        RTS
;
; STARTADRESSE
;
        *=  #02E0
        .WORD #5000
```

Star-Castle

Dieses Programm wurde auf einem Atari 800 geschrieben. Es müßte aber auf allen 8-Bit-Ataris mit 48 KByte laufen. Zum Spielen benötigt man einen Joystick. Die RESET-Taste ist auf Kaltstart programmiert.

8 Bit

Das Programm sollte zunächst abgespeichert und dann mit RUN gestartet werden. Nun dauert es ein wenig, bis das Titelbild erscheint. Anschließend sieht man das Star-Castle. Solange auf dem Bildschirm der Schriftzug PLEASE WAIT steht, wird das Maschinenprogramm in den Speicher gepoked. Erscheint links oben PRESS START, kann mit dem Spiel begonnen werden.

Die Aufgabe besteht nun darin, angreifende Roboter abzuwehren, die den Zentralcomputer durch Erschütterung zerstören wollen. Hierzu rammen sie beständig das Star-Castle. Wurde ein Computer zerstört, so kann durch Drücken des Feuerknopfes weitergespielt werden. Geschossen wird, indem man den Joystick nach rechts/links und oben/unten bewegt.

Das Spiel wird nach jedem Level schneller, der sich dann erhöht, wenn der Timer bei Null angekommen ist. Ist dies der Fall, kann das Spiel durch Druck auf den Feuerknopf fortgesetzt werden. Zunächst erhält man nach 9000 Punkten einen neuen Computer, später immer nach 10.000 Punkten. Die ESCAPE-Taste ist als Pausentaste programmiert. Durch Druck auf die SPACE-Taste kann man nach der Pause weiterspielen.

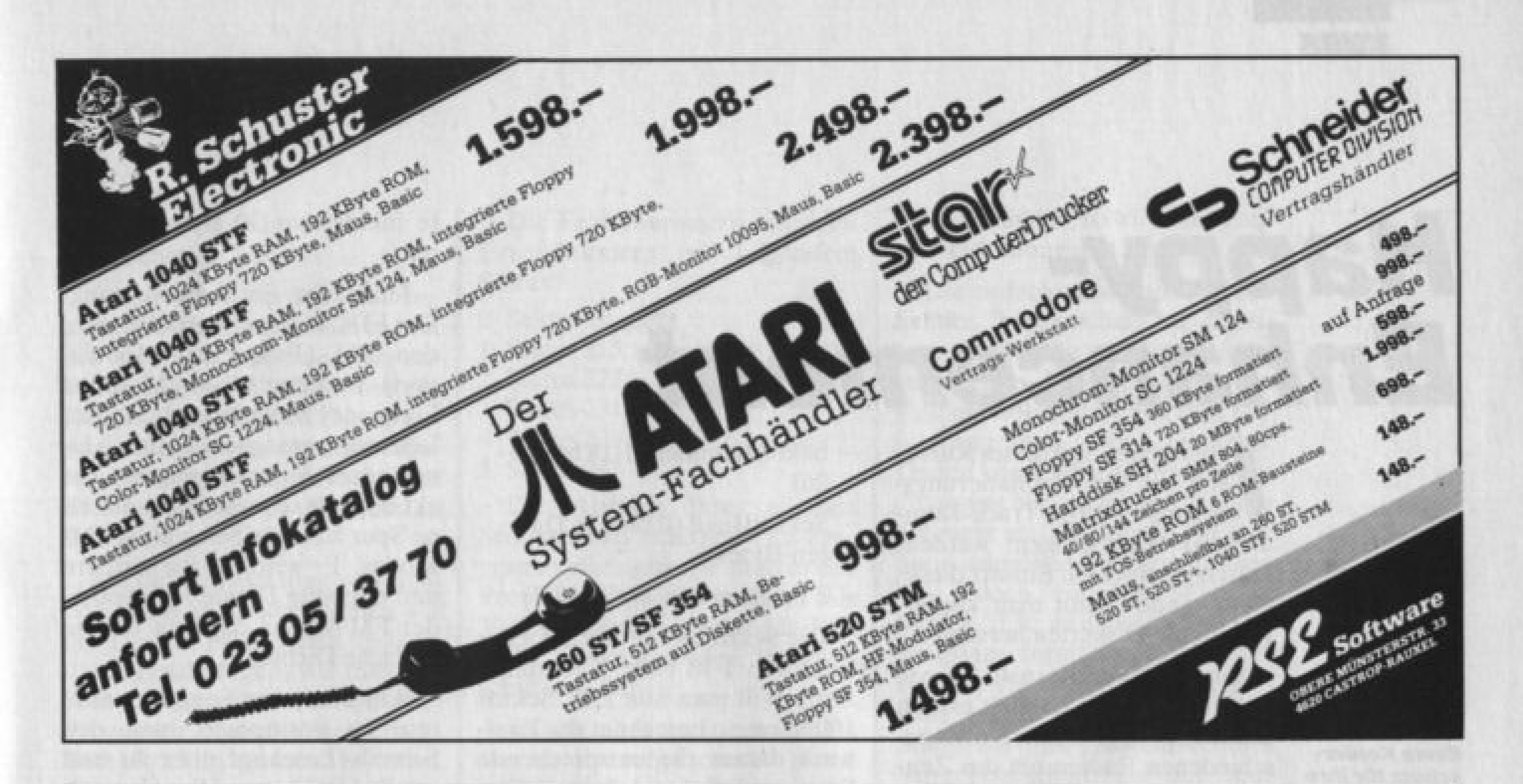
Dirk Jacobasch

Listing

```
10 REM ******************
11 REM *  STAR-CASTLE   *
12 REM *  (C) 1986 BY   *
13 REM * DIRK JACOBASCH *
14 REM * for ATARI 800  *
15 REM *     (XL/XE)    *
16 REM ******************
17 DIM HEX$(4):ADD=1536:A=0:Z=2:Y=110
100 GRAPHICS 8+16:SETCOLOR 2,0,0:POKE
709,0:COLOR 1
110 POKE 32835,2:POKE 32840,2
120 Y=Y+Z:Z=Z+2.31:IF Y>191 THEN 140
130 PLOT 0,Y:DRAWTO 319,Y:GOTO 120
140 X=160:Y=112:X1=X:Y1=191
150 PLOT X,Y:DRAWTO X1,Y1:PLOT X-1,Y:D
RAWTO X1-1,Y1
160 X1=X1-30:IF X1<1 THEN X=X-5:Z1=18:
GOTO 180
170 X=X-5:GOTO 150
180 Y1=Y1-Z1:IF Y1<115 THEN 210
185 PLOT X,Y:DRAWTO 0,Y1
190 PLOT X-1,Y:DRAWTO 0,Y-1:Z1=Z1-2.4:
X=X-10:IF Z1<3 THEN Z1=3
200 GOTO 180
210 X=160:Y=112:X1=X:Y1=191
220 PLOT X,Y:DRAWTO X1,Y1:PLOT X-1,Y:D
RAWTO X1-1,Y1
222 X1=X1+30:IF X1>318 THEN X=X+5:Z1=1
8:GOTO 240
230 X=X+5:GOTO 220
240 Y1=Y1-Z1:IF Y1<115 THEN 300
250 PLOT X,Y:DRAWTO 319,Y1:PLOT X+1,Y:
DRAWTO 319,Y1-1:Z1=Z1-2.4:X=X+10
260 IF Z1<3 THEN Z1=3
270 GOTO 240
300 FOR X=0 TO 100:PLOT INT(RND(0)*319
),INT(RND(0)*100):NEXT X
310 FOR Z=0 TO 20:X=INT(RND(0)*317)+1:
Y=INT(RND(0)*100):PLOT X,Y:DRAWTO X-1,
Y:NEXT Z
400 RESTORE 1000
405 FOR X=0 TO 42:POKE 33423+X,0:POKE
33623+X,0:NEXT X
410 FOR X=0 TO 12:READ A:POKE 33438+X,
A:NEXT X
420 RESTORE 1010
430 FOR X=0 TO 10:READ A:POKE 33639+X,
A:NEXT X
440 FOR X=0 TO 360 STEP 17:PLOT 160,90
:DRAWTO 160+COS(X)*7,90-SIN(X)*7
450 PLOT 60,54:DRAWTO 50+SIN(X)*4,50+C
OS(X)*4:NEXT X:Q=0
460 FOR X=0 TO 10:NEXT X:Q=Q+1:POKE 70
9,Q:IF Q=15 THEN 480
470 GOTO 460
480 FOR X=0 TO 1500:NEXT X
490 FOR X=0 TO 10:NEXT X:Q=Q-1:POKE 70
9,Q
500 IF Q=0 THEN 2000
510 GOTO 490
1000 DATA 36,14,42,14,00,48,50,37,51,3
7,46,52,51
1010 DATA 51,52,33,50,13,35,33,51,52,4
4,37
2000 GRAPHICS 7+16:POKE 36766,4:POKE 3
6767,4:POKE 36857,4
2005 FOR X=707 TO 712:POKE X,0:NEXT X
2010 FOR X=0 TO 90:PLOT INT(RND(0)*159
),INT(RND(0)*95):COLOR INT(RND(0)*3)+1
:NEXT X
2020 FOR X=37000 TO 37080:POKE X,0:NEX
T X:RESTORE 2060
2040 FOR X=37055 TO 37065:READ A:POKE
X,A:NEXT X:GOTO 2070
2060 DATA 48,44,37,33,51,37,0,55,33,41
,52
2070 RESTORE 2090
2075 FOR X=40640 TO 40680:POKE X,0:NEX
T X
2080 FOR X=40642 TO 40676:READ A:POKE
X,A:NEXT X:GOTO 3090
2090 DATA 44,33,51,52,0,51,35,47,50,37
,26,16,16,16,16,16,16,16,0
2100 DATA 0,0,0,44,33,51,52,0,44,37,54
,37,44,26,16,16
3090 COLOR 0:FOR X=67 TO 93:PLOT X,30:
DRAWTO X,54:NEXT X
3100 COLOR 3:PLOT 71,35:DRAWTO 88,35:D
RAWTO 88,49:DRAWTO 71,49:DRAWTO 71,35
3110 PLOT 71,35:DRAWTO 75,39:DRAWTO 84
,39:DRAWTO 88,35
3120 PLOT 75,39:DRAWTO 75,45:DRAWTO 71
,49
3130 PLOT 75,45:DRAWTO 84,45:DRAWTO 88
,49:PLOT 84,45
3140 DRAWTO 84,39:COLOR 2:PLOT 70,40:D
RAWTO 70,44
3150 PLOT 89,40:DRAWTO 89,44:PLOT 77,3
4:DRAWTO 83,34
```

```
3160 PLOT 77,50:DRAWTO 83,50:COLOR 3
3170 PLOT 80,51:DRAWTO 80,52:PLOT 80,3
3:DRAWTO 80,32
3180 PLOT 69,42:DRAWTO 68,42:PLOT 90,4
2:DRAWTO 91,42
3190 COLOR 1:PLOT 71,35:PLOT 88,35:PLO
T 88,49:PLOT 71,49
3195 A=192:B=128:C=160
3196 FOR X=100 TO 0 STEP -0.5:SOUND 3,
X,6,10:NEXT X
3197 SOUND 2,180,12,13:SOUND 3,181,12,
13
3210 A=A+1:B=B+1:C=C+1:D=D+10:FOR X=0
TO 20:NEXT X
3220 IF A=198 THEN A=197
3230 IF B=140 THEN GOTO 3890
3240 IF C=169 THEN C=168
3250 POKE 708,A:POKE 709,B:POKE 710,C
3260 GOTO 3210
3890 RESTORE 10000
3900 S=0
4000 FOR I=1 TO 8:READ HEX$
4010 IF HEX$="END" THEN 4300
4015 IF HEX$="ZZ" THEN ADD=31232:GOTO
4000
4017 IF HEX$="XX" THEN ADD=32768:GOTO
4000
4018 IF HEX$="YY" THEN ADD=29184:GOTO
4000
4019 IF HEX$="WW" THEN ADD=32000:GOTO
4000
4020 HI=ASC(HEX$(1,1))-48:IF HI>9 THEN
 HI=HI-7
4030 LO=ASC(HEX$(2,2))-48:IF LO>9 THEN
 LO=LO-7
4040 Z=HI*16+LO:S=S+Z
4050 POKE ADD,Z:ADD=ADD+1:NEXT I:GOTO
4000
4300 IF S<>364261 THEN GRAPHICS 0:SETC
OLOR 2,0,0:POSITION 13,10:? "DATA-ERRO
R -/":POKE 755,0:END
4400 X=USR(31232)
4500 GOTO 4500
10000 DATA 00,00,81,C3,24,3C,7E,5A
10001 DATA 7E,3C,42,C3,00,18,24,42
10002 DATA 99,FF,99,42,24,18,00,00
10003 DATA 18,24,3C,7E,03,7E,3C,24
10004 DATA 18,00,00,10,38,38,44,FE
10005 DATA 06,38,10,00,00,00,00,00
10006 DATA 30,32,25,33,33,00,33,34
10007 DATA 21,32,34,00,00,00,00,00
10008 DATA 33,34,21,32,0D,23,21,33
10009 DATA 34,2C,25,00,38,23,3D,00
10010 DATA 22,39,00,24,0E,2A,0E,00
10011 DATA 34,29,2D,25,32,1A,19,19
10012 DATA 19,19,00,00,00,00,00,00
10013 DATA 2C,25,36,25,2C,1A,10,10
10014 DATA 00,00,ZZ,A2,2E,A9,00,9D
10015 DATA 86,90,CA,D0,F8,A2,43,BD
10016 DATA 2F,06,9D,87,90,CA,D0,F7
10017 DATA A9,FF,8D,44,02,AD,1F,D0
10018 DATA C9,06,D0,F9,A9,33,8D,08
10019 DATA 90,A9,23,8D,09,90,A9,1A
10020 DATA 8D,8A,90,A9,10,A2,07,9D
10021 DATA 8A,90,CA,D0,FA,A9,23,8D
10022 DATA 93,90,A9,1A,8D,94,90,A9
10023 DATA 13,8D,95,90,A9,73,8D,07
10024 DATA D4,A9,DC,8D,C0,02,A9,1B
10025 DATA 8D,C1,02,A9,37,8D,C2,02
10026 DATA A9,88,8D,C3,02,A2,06,A9
10027 DATA 00,9D,00,D0,CA,D0,FA,A9
10028 DATA 3E,8D,2F,02,A9,05,8D,0C
10029 DATA D0,A9,03,8D,1D,D0,A9,00
10030 DATA 8D,00,7E,8D,01,7E,A9,8A
10031 DATA 8D,02,7E,A9,72,8D,03,7E
10032 DATA A2,03,A9,00,9D,04,7E,CA
10033 DATA D0,FA,A2,04,A9,0A,9D,0A
10034 DATA 7E,CA,D0,FA,A9,00,8D,0E
10035 DATA 7E,8D,10,7E,A9,08,8D,0F
10036 DATA 7E,A9,6E,8D,11,7E,A9,94
10037 DATA 8D,12,7E,A9,00,8D,13,7E
10038 DATA 8D,18,7E,A9,80,8D,14,7E
10039 DATA 8D,16,7E,8D,17,7E,8D,19
10040 DATA 7E,A9,FF,8D,15,7E,A2,0E
10041 DATA A9,FF,9D,19,7E,CA,D0,FA
10042 DATA A2,03,A9,00,9D,26,7E,CA
10043 DATA D0,FA,8D,92,90,A9,0A,8D
10044 DATA 08,7E,A9,0B,8D,0C,7E,8D
10045 DATA 0D,7E,A9,FF,8D,2B,7E,8D
10046 DATA 2C,7E,8D,2D,7E,8D,2E,7E
10047 DATA A9,00,8D,2F,7E,8D,30,7E
10048 DATA 8D,31,7E,8D,32,7E,8D,37
10049 DATA 7E,A9,60,8D,33,7E,8D,34
10050 DATA 7E,8D,35,7E,8D,36,7E,8D
10051 DATA 38,7E,A9,80,8D,39,7E,A9
10052 DATA 06,8D,3A,7E,8D,3B,7E,A9
10053 DATA 00,8D,3C,7E,8D,3D,7E,8D
10054 DATA 41,7E,A9,FF,8D,40,7E,A9
10055 DATA 04,8D,3F,7E,A2,FF,A9,00
10056 DATA 9D,00,73,9D,00,74,9D,00
10057 DATA 75,9D,00,76,9D,00,77,CA
10058 DATA D0,EE,8D,09,7E,A2,10,9D
10059 DATA CA,90,CA,D0,FA,4C,00,80
10060 DATA 00,00,00,00,00,00,00,XX
10061 DATA AD,78,02,C9,07,D0,05,A9
10062 DATA 01,8D,04,7E,AD,78,02,C9
10063 DATA 0B,D0,05,A9,01,8D,05,7E
10064 DATA AD,78,02,C9,0E,D0,05,A9
10065 DATA 01,8D,06,7E,AD,78,02,C9
10066 DATA 0D,D0,05,A9,01,8D,07,7E
10067 DATA AD,04,7E,C9,01,D0,2B,A9
10068 DATA 01,8D,0E,7E,AE,0A,7E,CA
10069 DATA 8E,0A,7E,D0,1D,A9,0C,8D
10070 DATA 0A,7E,AE,02,7E,E8,8E,02
10071 DATA 7E,8E,04,D0,E0,F0,D0,0A
10072 DATA A9,8A,8D,02,7E,A9,00,8D
10073 DATA 04,7E,AD,05,7E,C9,01,D0
10074 DATA 2B,A9,01,8D,0E,7E,AE,0B
10075 DATA 7E,CA,8E,0B,7E,D0,1D,A9
10076 DATA 0C,8D,0B,7E,AE,03,7E,CA
10077 DATA 8E,03,7E,8E,05,D0,E0,10
10078 DATA D0,0A,A9,73,8D,03,7E,A9
10079 DATA 00,8D,05,7E,AD,06,7E,C9
10080 DATA 01,D0,2D,A9,01,8D,0E,7E
10081 DATA AE,0C,7E,CA,8E,0C,7E,D0
10082 DATA 1F,A9,0B,8D,0C,7E,AE,11
10083 DATA 7E,CA,8E,11,7E,A9,10,9D
10084 DATA 00,73,E0,02,D0,0A,A9,6E
10085 DATA 8D,11,7E,A9,00,8D,06,7E
10086 DATA AD,07,7E,C9,01,D0,2D,A9
10087 DATA 01,8D,0E,7E,AE,0D,7E,CA
10088 DATA 8E,0D,7E,D0,1F,A9,0B,8D
10089 DATA 0D,7E,AE,12,7E,E8,8E,12
10090 DATA 7E,A9,40,9D,00,73,E0,FD
10091 DATA D0,0A,A9,94,8D,12,7E,A9
10092 DATA 00,8D,07,7E,A9,C5,8D,C4
10093 DATA 02,A9,8A,8D,C5,02,A9,A8
10094 DATA 8D,C6,02,AD,0E,7E,C9,01
10095 DATA D0,2E,AE,0F,7E,CA,8E,0F
10096 DATA 7E,D0,25,A9,08,8D,0F,7E
10097 DATA A9,A5,8D,01,D2,AE,10,7E
10098 DATA E8,8E,10,7E,8E,00,D2,D0
10099 DATA 0F,A9,00,8D,0E,7E,8D,10
10100 DATA 7E,8D,01,D2,A9,08,0F,7E
10101 DATA AE,11,7E,A9,00,E8,E8,E8
10102 DATA E8,E8,9D,00,73,AE,12,7E
10103 DATA CA,CA,CA,CA,CA,9D,00,73
10104 DATA A9,7F,8D,06,D0,8D,07,D0
10105 DATA AD,0A,D2,8D,DB,96,AD,0A
10106 DATA D2,8D,DC,96,AD,0A,D2,8D
10107 DATA 03,97,AD,0A,D2,8D,04,97
10108 DATA AD,0A,D2,8D,B3,96,AD,0A
10109 DATA D2,8D,B4,96,AD,0A,D2,8D
10110 DATA 2B,97,AD,0A,D2,8D,2C,97
10111 DATA AD,0A,D2,8D,53,97,AD,0A
10112 DATA D2,8D,54,97,AE,1B,7E,CA
10113 DATA 8E,1B,7E,D0,10,AD,1C,7E
10114 DATA 8D,1B,7E,AE,13,7E,E8,8E
10115 DATA 13,7E,8E,00,D0,AE,1D,7E
10116 DATA CA,8E,1D,7E,D0,51,A9,2C
10117 DATA 8D,1D,7E,AD,27,7E,C9,00
10118 DATA D0,1F,AE,14,7E,E8,8E,14
10119 DATA 7E,A0,0C,B9,00,06,9D,00
10120 DATA 74,CA,88,D0,F6,AD,14,7E
10121 DATA C9,96,D0,05,A9,01,8D,27
```

```
10122 DATA 7E,AD,27,7E,C9,01,D0,1F
10123 DATA AE,14,7E,CA,8E,14,7E,A0
10124 DATA 0C,B9,00,06,9D,00,74,CA
10125 DATA 88,D0,F6,AD,14,7E,C9,73
10126 DATA D0,05,A9,00,8D,27,7E,EA
10127 DATA AE,1E,7E,CA,8E,1E,7E,D0
10128 DATA 10,AD,1F,7E,8D,1E,7E,AE
10129 DATA 15,7E,CA,8E,15,7E,8E,01
10130 DATA D0,AE,20,7E,CA,8E,20,7E
10131 DATA D0,51,A9,2C,8D,20,7E,AD
10132 DATA 28,7E,C9,00,D0,1F,AE,16
10133 DATA 7E,E8,8E,16,7E,A0,0C,B9
10134 DATA 0B,06,9D,00,75,CA,88,D0
10135 DATA F6,AD,16,7E,C9,95,D0,05
10136 DATA A9,01,8D,28,7E,AD,28,7E
10137 DATA C9,01,D0,1F,AE,16,7E,CA
10138 DATA 8E,16,7E,A0,0C,B9,0B,06
10139 DATA 9D,00,75,CA,88,D0,F6,AD
10140 DATA 16,7E,C9,76,D0,05,A9,00
10141 DATA 8D,28,7E,AE,21,7E,CA,8E
10142 DATA 21,7E,D0,39,A9,30,8D,21
10143 DATA 7E,AD,29,7E,C9,00,D0,13
10144 DATA AE,17,7E,E8,8E,17,7E,8E
10145 DATA 02,D0,E0,88,D0,05,A9,01
10146 DATA 8D,29,7E,AD,29,7E,C9,01
10147 DATA D0,13,AE,17,7E,CA,8E,17
10148 DATA 7E,8E,02,D0,E0,71,D0,05
10149 DATA A9,00,8D,29,7E,AE,22,7E
10150 DATA CA,8E,22,7E,D0,19,AD,23
10151 DATA 7E,8D,22,7E,AE,18,7E,E8
10152 DATA 8E,18,7E,A0,0C,B9,15,06
10153 DATA 9D,00,76,CA,88,D0,F6,AE
10154 DATA 24,7E,CA,8E,24,7E,D0,39
10155 DATA A9,30,8D,24,7E,AD,2A,7E
10156 DATA C9,00,D0,13,AE,19,7E,E8
10157 DATA 8E,19,7E,8E,03,D0,E0,8A
10158 DATA D0,05,A9,01,8D,2A,7E,AD
10159 DATA 2A,7E,C9,01,D0,13,AE,19
10160 DATA 7E,CA,8E,19,7E,8E,03,D0
10161 DATA E0,72,D0,05,A9,00,8D,2A
10162 DATA 7E,AE,25,7E,CA,8E,25,7E
10163 DATA D0,19,AD,26,7E,8D,25,7E
10164 DATA AE,1A,7E,CA,8E,1A,7E,A0
10165 DATA 0C,B9,21,06,9D,00,77,CA
10166 DATA 88,D0,F6,AE,08,7E,CA,8E
10167 DATA 08,7E,D0,45,A9,0A,8D,08
10168 DATA 7E,AE,B9,90,CA,8E,B9,90
10169 DATA E0,0F,D0,35,A9,19,8D,B9
10170 DATA 90,AE,B8,90,CA,8E,B8,90
10171 DATA E0,0F,D0,25,A9,19,8D,B8
10172 DATA 90,AE,B7,90,CA,8E,B7,90
10173 DATA E0,0F,D0,15,A9,19,8D,B7
10174 DATA 90,AE,B6,90,CA,8E,B6,90
10175 DATA E0,0F,D0,05,A9,19,8D,B6
10176 DATA 90,A2,04,BD,B5,90,C9,10
10177 DATA D0,08,CA,D0,F6,A9,01,8D
10178 DATA 09,7E,AD,09,7E,C9,01,D0
10179 DATA 7A,A2,FF,A9,00,9D,00,73
10180 DATA 9D,00,74,9D,00,75,9D,00
10181 DATA 76,9D,00,77,CA,D0,EE,AE
10182 DATA C7,90,E8,8E,C7,90,E0,1A
10183 DATA D0,15,A9,10,8D,C7,90,AE
10184 DATA C6,90,E8,8E,C6,90,E0,1A
10185 DATA D0,05,A9,10,8D,C6,90,A9
10186 DATA 00,8D,00,7E,8D,01,7E,A9
10187 DATA 88,8D,02,7E,A9,72,8D,03
10188 DATA 7E,A9,67,8D,11,7E,A9,8F
10189 DATA 8D,12,7E,A9,00,8D,13,7E
10190 DATA 8D,15,7E,8D,18,7E,8D,1A
10191 DATA 7E,8D,03,D2,8D,37,7E,8D
10192 DATA 38,7E,8D,0E,7E,8D,10,7E
10193 DATA A9,08,8D,0F,7E,8D,3A,7E
10194 DATA 8D,01,D2,AD,09,7E,C9,01
10195 DATA D0,61,EA,A9,00,8D,01,D2
10196 DATA 8D,03,D2,AD,84,02,C9,00
10197 DATA D0,F9,A9,19,8D,B9,90,8D
10198 DATA B8,90,8D,B7,90,8D,B6,90
10199 DATA A9,00,8D,09,7E,AD,1C,7E
10200 DATA 38,E9,0A,8D,1C,7E,AD,1F
10201 DATA 7E,38,E9,0A,8D,1F,7E,AD
10202 DATA 23,7E,38,E9,0A,8D,23,7E
10203 DATA AD,26,7E,38,E9,0A,8D,26
10204 DATA 7E,A9,00,8D,00,D0,8D,01
10205 DATA D0,8D,04,7E,8D,05,7E,8D
10206 DATA 06,7E,8D,07,7E,8D,04,D0
10207 DATA 8D,05,D0,AD,1C,7E,C9,20
10208 DATA B0,05,A9,25,8D,1C,7E,AD
10209 DATA 1F,7E,C9,20,B0,05,A9,25
10210 DATA 8D,1F,7E,AD,23,7E,C9,20
10211 DATA B0,05,A9,25,8D,23,7E,AD
10212 DATA 26,7E,C9,20,B0,05,A9,25
10213 DATA 8D,26,7E,AD,FC,02,C9,1C
10214 DATA D0,07,AD,FC,02,C9,21,D0
10215 DATA F9,EA,AD,08,D0,C9,02,D0
10216 DATA 08,A9,01,8D,2F,7E,8D,1E
10217 DATA D0,AD,09,D0,C9,01,D0,08
10218 DATA A9,01,8D,30,7E,8D,1E,D0
10219 DATA AD,0A,D0,C9,04,D0,08,A9
10220 DATA 01,8D,31,7E,8D,1E,D0,AD
10221 DATA 0B,D0,C9,08,D0,08,A9,01
10222 DATA 8D,32,7E,8D,1E,D0,AD,2F
10223 DATA 7E,C9,01,D0,42,A2,0D,AC
10224 DATA 16,7E,AD,0A,D2,99,00,75
10225 DATA 88,CA,D0,F6,A9,01,8D,37
10226 DATA 7E,A9,FF,8D,20,7E,AE,33
10227 DATA 7E,CA,8E,33,7E,D0,20,A9
10228 DATA 60,8D,33,7E,A9,00,8D,15
10229 DATA 7E,8D,01,D0,8D,2F,7E,8D
10230 DATA 04,7E,8D,04,D0,A9,88,8D
10231 DATA 02,7E,A9,01,8D,2C,7E,AD
10232 DATA 30,7E,C9,01,D0,42,A2,0D
10233 DATA AC,14,7E,AD,0A,D2,99,00
10234 DATA 74,88,CA,D0,F6,A9,01,8D
10235 DATA 37,7E,A9,FF,8D,1D,7E,AE
10236 DATA 34,7E,CA,8E,34,7E,D0,20
10237 DATA A9,60,8D,34,7E,A9,00,8D
10238 DATA 13,7E,8D,00,D0,8D,30,7E
10239 DATA 8D,05,7E,8D,05,D0,A9,72
10240 DATA 8D,03,7E,A9,01,8D,2C,7E
10241 DATA AD,31,7E,C9,01,D0,49,A2
10242 DATA 0C,AC,18,7E,AD,0A,D2,99
10243 DATA 00,76,88,CA,D0,F6,A9,01
10244 DATA 8D,37,7E,A9,FF,8D,22,7E
10245 DATA AE,35,7E,CA,8E,35,7E,D0
10246 DATA 27,A9,60,8D,35,7E,A9,00
10247 DATA 8D,18,7E,8D,31,7E,8D,06
10248 DATA 7E,A9,6E,8D,11,7E,A2,00
10249 DATA A9,00,9D,00,73,9D,00,76
10250 DATA CA,D0,F7,A9,01,8D,2C,7E
10251 DATA AD,32,7E,C9,01,D0,49,A2
10252 DATA 0C,AC,1A,7E,AD,0A,D2,99
10253 DATA 00,77,88,CA,D0,F6,A9,01
10254 DATA 8D,37,7E,A9,FF,8D,25,7E
10255 DATA AE,36,7E,CA,8E,36,7E,D0
10256 DATA 27,A9,60,8D,36,7E,A9,00
10257 DATA 8D,1A,7E,8D,32,7E,8D,07
10258 DATA 7E,A9,94,8D,12,7E,A2,00
10259 DATA A9,00,9D,00,73,9D,00,77
10260 DATA CA,D0,F7,A9,01,8D,2C,7E
10261 DATA AD,37,7E,C9,01,D0,2A,AE
10262 DATA 3A,7E,CA,8E,3A,7E,D0,1E
10263 DATA A9,04,8D,3A,7E,A9,CE,8D
10264 DATA 03,D2,AE,38,7E,E8,8E,02
10265 DATA D2,8E,38,7E,D0,0B,A9,00
10266 DATA 8D,38,7E,8D,03,D2,8D,37
10267 DATA 7E,AE,39,7E,CA,8E,39,7E
10268 DATA D0,25,A9,E0,8D,39,7E,AD
10269 DATA 1C,06,0A,8D,1C,06,C9,C0
10270 DATA D0,05,A9,06,8D,1C,06,AD
10271 DATA 28,06,0A,8D,28,06,C9,00
10272 DATA D0,05,A9,06,8D,28,06,AE
10273 DATA 2B,7E,CA,8E,2B,7E,D0,75
10274 DATA A9,FF,8D,2B,7E,AE,91,90
10275 DATA E8,8E,91,90,E0,1A,D0,65
10276 DATA A9,10,8D,91,90,AE,90,90
10277 DATA E8,8E,90,90,E0,1A,D0,55
10278 DATA A9,10,8D,90,90,AE,8F,90
10279 DATA E8,8E,8F,90,E0,1A,D0,45
10280 DATA A9,10,8D,8F,90,AE,8E,90
10281 DATA E8,8E,8E,90,E0,1A,D0,35
10282 DATA A9,10,8D,8E,90,AE,8D,90
10283 DATA E8,8E,8D,90,E0,1A,D0,25
10284 DATA A9,10,8D,8D,90,AE,8C,90
10285 DATA E8,8E,8C,90,E0,1A,D0,15
10286 DATA A9,10,8D,8C,90,AE,8B,90
```




```
10287 DATA E8,8E,8B,90,E0,1A,D0,05
10288 DATA A9,10,8D,8C,90,AD,2C,7E
10289 DATA C9,01,D0,55,AE,8F,90,E8
10290 DATA 8E,8F,90,E0,1A,D0,45,A9
10291 DATA 10,8D,8F,90,AE,8E,90,E8
10292 DATA 8E,8E,90,E0,1A,D0,35,A9
10293 DATA 10,8D,8E,90,AE,8D,90,E8
10294 DATA 8E,8D,90,E0,1A,D0,25,A9
10295 DATA 10,8D,8E,90,AE,8C,90,E8
10296 DATA 8E,8C,90,E0,1A,D0,15,A9
10297 DATA 10,8D,8C,90,AE,8B,90,E8
10298 DATA 8E,8B,90,E0,1A,D0,05,A9
10299 DATA 10,8D,8B,90,A9,00,8D,2C
10300 DATA 7E,AD,2D,7E,C9,FF,D0,1A
10301 DATA AD,8E,90,C9,19,D0,13,AD
10302 DATA 95,90,C9,19,F0,0C,AE,95
10303 DATA 90,E8,8E,95,90,A9,01,8D
10304 DATA 2D,7E,AD,8E,90,C9,10,D0
10305 DATA 05,A9,FF,8D,2D,7E,AE,3B
10306 DATA 7E,CA,8E,3B,7E,D0,35,A9
10307 DATA FF,8D,3B,7E,AE,3C,7E,E8
10308 DATA 8E,3C,7E,E0,01,D0,0F,A9
10309 DATA 18,8D,02,06,A9,3C,8D,03
10310 DATA 06,A9,10,8D,23,06,E0,02
10311 DATA D0,12,A9,81,8D,02,06,A9
10312 DATA C3,8D,03,06,A9,00,8D,3C
10313 DATA 7E,8D,23,06,AD,13,7E,C9
10314 DATA 71,D0,05,A9,01,8D,3D,7E
10315 DATA AD,15,7E,C9,87,D0,05,A9
10316 DATA 01,8D,3D,7E,AD,18,7E,C9
10317 DATA 75,D0,05,A9,01,8D,3D,7E
10318 DATA AD,1A,7E,C9,95,D0,05,A9
10319 DATA 01,8D,3D,7E,AD,3D,7E,C9
10320 DATA 01,D0,03,4C,00,72,A9,0F
10321 DATA 8D,80,73,8D,81,73,A9,C5
10322 DATA 8D,05,D2,8D,07,D2,A9,B4
10323 DATA 8D,04,D2,A9,B5,8D,06,D2
10324 DATA 8D,1E,D0,A9,00,8D,4D,00
10325 DATA AD,13,7E,C9,05,D0,35,AD
10326 DATA 0A,D2,29,03,C9,00,D0,08
10327 DATA A9,00,8D,D4,81,8D,FA,81
10328 DATA C9,01,D0,08,A9,0B,8D,D4
10329 DATA 81,8D,FA,81,C9,02,D0,08
10330 DATA A9,15,8D,D4,81,8D,FA,81
10331 DATA C9,03,D0,08,A9,21,8D,D4
10332 DATA 81,8D,FA,81,AD,15,7E,C9
10333 DATA FD,D0,35,AD,0A,D2,29,03
10334 DATA C9,00,D0,08,A9,00,8D,48
10335 DATA 82,8D,6E,82,C9,01,D0,08
10336 DATA A9,0B,8D,48,82,8D,6E,82
10337 DATA C9,02,D0,08,A9,15,8D,48
10338 DATA 82,8D,6E,82,C9,03,D0,08
10339 DATA A9,21,8D,48,82,8D,6E,82
10340 DATA AD,18,7E,C9,04,D0,29,AD
10341 DATA 0A,D2,29,03,C9,00,D0,05
10342 DATA A9,00,8D,DE,82,C9,01,D0
10343 DATA 05,A9,0B,8D,DE,82,C9,02
10344 DATA D0,05,A9,15,8D,DE,82,C9
10345 DATA 03,D0,05,A9,21,8D,DE,82
10346 DATA AD,1A,7E,C9,FF,D0,29,AD
10347 DATA 0A,D2,29,03,C9,00,D0,05
10348 DATA A9,00,8D,42,83,C9,01,D0
10349 DATA 05,A9,0B,8D,42,83,C9,02
10350 DATA D0,05,A9,15,8D,42,83,C9
10351 DATA 03,D0,05,A9,21,8D,42,83
10352 DATA 4C,00,80,00,00,00,00,00
10353 DATA 00,00,YY,A9,8E,8D,01,D2
10354 DATA A2,FF,A9,00,9D,00,73,9D
10355 DATA 00,74,9D,00,75,9D,00,76
10356 DATA 9D,00,77,CA,D0,EC,8D,13
10357 DATA 7E,8D,15,7E,8D,18,7E,8D
10358 DATA 1A,7E,8D,00,D0,8D,01,D0
10359 DATA 8D,03,D2,8D,04,7E,8D,05
10360 DATA 7E,8D,06,7E,8D,07,7E,AE
10361 DATA 3F,7E,CA,8E,3F,7E,E0,00
10362 DATA F0,2E,AE,40,7E,CA,8E,40
10363 DATA 7E,D0,F7,A9,FF,8D,40,7E
10364 DATA AD,0A,D2,8D,C8,02,AE,41
10365 DATA 7E,E8,8E,41,7E,8E,00,D2
10366 DATA E0,40,D0,DE,A9,00,8D,41
10367 DATA 7E,AD,3F,7E,C9,00,D0,C7
10368 DATA A9,04,8D,3F,7E,A9,00,8D
10369 DATA 01,D2,8D,DB,96,8D,DC,96
10370 DATA 8D,B3,96,8D,B4,96,8D,03
10371 DATA 97,8D,04,97,8D,2B,97,8D
10372 DATA 2C,97,8D,53,97,8D,54,97
10373 DATA AE,95,90,CA,8E,95,90,E0
10374 DATA 10,D0,03,4C,00,7D,A9,00
10375 DATA 8D,3D,7E,8D,C8,02,8D,4D
10376 DATA 00,AD,84,02,C9,00,D0,F6
10377 DATA A9,8A,8D,02,7E,A9,73,8D
10378 DATA 03,7E,A9,6E,8D,11,7E,A9
10379 DATA 94,8D,12,7E,A9,00,8D,04
10380 DATA D0,8D,05,D0,8D,0E,7E,8D
10381 DATA 10,7E,8D,37,7E,8D,38,7E
10382 DATA 8D,04,7E,8D,05,7E,8D,06
10383 DATA 7E,8D,07,7E,A9,08,8D,0F
10384 DATA 7E,8D,3A,7E,4C,00,80,WW
10385 DATA A2,FF,A9,00,9D,00,73,9D
10386 DATA 00,74,9D,00,75,9D,00,76
10387 DATA 9D,00,77,CA,D0,EE,A9,CD
10388 DATA 8D,05,D2,8D,07,D2,A2,09
10389 DATA BD,4A,7D,9D,CB,90,CA,D0
10390 DATA F7,A9,00,8D,C8,02,A2,07
10391 DATA BD,8A,90,9D,CC,9E,CA,D0
10392 DATA F7,AD,C7,90,8D,E4,9E,AD
10393 DATA C6,90,8D,E3,9E,4C,00,7A
10394 DATA 00,00,00,27,21,2D,25,00
10395 DATA 2F,36,25,32,00,00,END
```

Happy-Enhancement

Einen Kopierschutz für Ihre Programme können Sie selbst programmieren. Selbst die Happy-Erweiterung ist da machtlos.

Im zweiten Teil dieses Kurses soll auf den Formatierungsvorgang (Write-Track-Kommando) eingegangen werden. Durch geschickten Einsatz dieses Kommandos kann man kopiergeschützte Disketten herstellen.

Eine Diskette ist in 40 Spuren unterteilt. Dabei handelt es sich um ringförmige Bahnen mit verschiedenen Radien um das Zentrum der Diskette. Der Schreib-/Lesekopf läßt sich über jede dieser Spuren exakt positionieren. Wenn sich nun die Diskette dreht, können die Daten einer Spur nacheinander gelesen werden.

8 Bit

Eine Spur beginnt mit der sogenannten Index Mark ($fc). Ihr folgen mehrere Gapbytes. (Sie tragen keinerlei Information und werden nur als Füll-Bytes zwischen den einzelnen Blöcken einer Spur benutzt). Dieser Gapbyte-Block wird als Post Index Mark bezeichnet.

Jede Spur enthält in der Regel 18 (bei einfacher Dichte) oder 26 (bei mittlerer Dichte) Sektoren, die aus zwei Teilen bestehen, dem Header und dem Datenfeld. Vor einem Header liegt ein weiterer Gapbyte-Block (Pre Address Mark). Der nun folgende Header ist für die Unterscheidung der Sektoren einer Spur zuständig. Er wird für jeden Sektor während des Formatierens angelegt und enthält 7 Byte, die folgende Bedeutung haben:

- Address Mark ($fe)
- Spurnummer (0-39)
- Seitennummer (0 oder 1)
- Sektornummer (1-18 bzw. 1-26)
- Sektorlänge (0 bis 128 Daten-Bytes)
- 2 Prüfsummen-Bytes ($f7)

Die Sektoren werden auf jeder Spur von 1-18 bzw. 1-26 numeriert. Will man nun z.B. Sektor 100 lesen, so berechnet das Laufwerk daraus die entsprechende Spur und Sektornummer (bei einfacher Dichte: Spur 5, Nummer 10; bei mittlerer Dichte: Spur 3, Nummer 22).

Die Berechnungsformel lautet:

Sektor = Spur*18 (26) + Sektornummer

Auf einen Header folgt zunächst ein Gapbyte-Block (Post Address Mark), dann erst das zugehörige Datenfeld. Es besitzt folgenden Aufbau:

- Data Address Mark ($fb)
- 128 Daten-Bytes
- 2 Prüfsummen-Bytes ($f7)

Nach den Prüfsummen-Bytes findet man wiederum Gapbytes (Post Data Mark). Danach folgt der nächste Sektor, der mit dem Gapbyte-Block (Pre Address Mark) beginnt. Ist der letzte Sektor formatiert, so wird der Rest der Spur mit Gapbytes aufgefüllt.

Um zu vermeiden, daß sich der erste und der letzte Sektor einer Spur überlappen, befindet sich zwischen beiden wieder eine Reihe von Gapbytes. Hier lassen sich weitere Sektoren unterbringen. Um Platz für noch mehr zu schaffen, kann man die einzelnen Gapbyte-Blöcke auf ein Minimum von einem Byte kürzen. Der Block zwischen Header und Datenfeld (Post Address Mark) muß jedoch in einfacher Dichte mindestens 15, in mittlerer Dichte mindestens 30 Byte beinhalten.

Jetzt zum Formatiervorgang. Der FDC hat zwei Betriebsarten, den FM (Frequency Modulation)- und den MFM (Modified Frequency Modulation)- Modus. Letzterer ermöglicht es durch eine andere Aufzeichnungsart, exakt doppelt so viele Bytes auf eine Spur zu schreiben wie im FM-Modus. Er wird für die mittlere und doppelte Dichte verwendet, der FM-Modus dagegen für die einfache Dichte.

Will man eine Spur formatieren, so positioniert man den Schreib-/Lesekopf über ihr und erteilt dem Floppy-Disc-Controller (FDC) das Write-Track-Kommando. Sobald der FDC einen Indeximpuls erhält, beginnt er, ein Byte nach dem anderen anzufordern, und schreibt es direkt auf Diskette. Tritt der nächste Indeximpuls eine Umdrehung später auf, so wird das Kommando beendet.

Das Indexloch der Disketten wird von einer 1050-Diskettenstation nicht zur Erzeugung eines Indeximpulses genutzt. Um diesen trotzdem an den FDC zu liefern, kommt ein Timer zum Einsatz, der – sobald er abgelaufen ist – einen Indeximpuls an den FDC sendet. Für das Formatieren einer Spur wird der Timer auf 210 ms gesetzt. Da die normale Umdrehungszeit 208 ms beträgt, sind dies 2 ms mehr als nötig. In dieser Zeit werden Gapbytes auf die Diskette geschrieben, um sicherzustellen, daß nichts von der vorherigen Formatierung auf dieser Spur zurückbleibt. Beim Formatieren kann man jedoch nicht jedes beliebige Byte auf die Diskette schreiben. Eine Reihe von Bytes ($f5-$ff) haben für den FDC eine besondere Bedeutung:

$f7 Schreibe 2 Prüfsummen-Bytes
$fb Data Address Mark
$fc Index Mark
$fe Address Mark

Im MFM-Modus kommen außerdem hinzu:

$f5 Schreibe $a1
$f6 Schreibe $c2

Diese Bytes werden dazu benutzt, bestimmte Markierungen, die sich von normalen Daten-Bytes abgrenzen lassen, auf die Diskette zu schreiben. Sie unterscheiden sich von normalen Daten-Bytes dadurch, daß bei ihnen der Taktimpuls zu bestimmten Bits fehlt.

Mit der Entwicklung von eigenen Schutzmethoden wollen wir die Theorie beschließen und uns damit beschäftigen, wie man mit dem beiliegenden Turbo-Basic-Programm diesen Write-Track-Befehl richtig einsetzt. Mit dem genannten Programm ist es möglich, Disketten so zu schützen, daß sie auch mit der Happy-Software nicht kopiert werden können. Die drei Assembler-Listings müssen zum Funktionieren des Basic-Programms nicht abgeschrieben werden; sie sind in dessen DATA-Zeilen bereits enthalten. Für den fortgeschrittenen Anwender wird jedoch Listing 2 interessant sein. Dieses Maschinenspracheprogramm läuft in der Diskettenstation ab und formatiert dort eine Spur.

Ich möchte nun auf drei verschiedene Schutzmethoden eingehen, die leicht zu realisieren sind:

1. Schutz durch Error-Sektoren
2. Schutz durch doppelte Sektoren
3. Schutz duch die Sektoranordnung

Das Programm beeinhaltet einen Beispielschutz, der alle drei Methoden zur Anwendung bringt. Formatieren Sie Spur 37 einer Diskette (die zuvor in mittlerer Dichte formatiert wurde) mit dem Beispielformat neu. Sie können diesen Schutz danach im Programm abfragen.

Wollen Sie selbst einen Schutz entwerfen, so müssen Sie zuerst im Format-Editor ein eigenes Format erstellen. Dabei ist zu beachten, daß die Fehlernummer nur von 0-5 und der Sektorinhalt nur von $b-$ff reichen darf.

Die Fehlernummern erzeugen Error-Sektoren mit folgendem Status:

0 Sektor ist ganz
1 Status 215
2 Status 223
3 Status 231
4 Status 239
5 Status 247

Die Abfrage Ihres eigenen Schutzes läßt sich direkt ins Programm einfügen. Sie muß in der Prozedur EIGABF stehen. Als Vorbild kann die Prozedur BEJABF (Abfrage des Beispielschutzes) dienen.

Es gibt aber auch den Schutz durch Error-Sektoren. Einen zerstörten Sektor testet man dadurch an, daß man versucht, ihn zu lesen; es ergibt sich dann ein Error 144. Nun sind jedoch nicht alle Error-Sektoren gleich. Eine direkt nach dem Leseversuch erfolgende Statusabfrage gibt Aufschluß über die Art der Zerstörung. Wenn Sie einen Error-Sektor abfragen, sollten Sie sich unbedingt auch über dessen Status informieren. Am besten wählen Sie drei verschiedene Error-Sektoren mit dem Status 215, 223 und 247. Solche mit 231 und 239 brauchen für die Abfrage wesentlich mehr Zeit.

Beim Schutz durch doppelte Sektoren definieren Sie ein Format, bei dem eine Sektornummer doppelt vorhanden ist, die Sektoren jedoch unterschiedliche Daten beinhalten. Um die Abfrage zu erleichtern, ist es günstig, wenn die doppelten Sektoren sich auf der Diskette etwa gegenüberliegen. Bei zweimaligem schnellen Lesen müssen die Daten voneinander abweichen.

Eine weitere Möglichkeit ist der Schutz durch die Sektoranordnung. Hier messen Sie die Zeit, die die Diskettenstation benötigt, um zwei Sektoren zu lesen. Durch deren Anordnung auf einer Spur kann diese Zeit beeinflußt werden. Liegen die beiden Sektoren sich auf der Diskette gegenüber, so hat der Lesevorgang etwa die Dauer einer halben Umdrehung. Liegen sie direkt nebeneinander, so wird eine ganze Umdrehungszeit benötigt.

Die einfachste Methode, einen Schutz "happysicher" zu gestalten, ist, ihn für mittlere Dichte vorzusehen. Dafür existiert – soweit mir bekannt – noch kein Kopierprogramm. In einfacher Dichte können Sie Ihr Programm "happysicher" schützen, indem Sie mehr als 20 Sektoren auf eine Spur schreiben. Hierzu müssen Sie jedoch einzelne Sektoren kürzen, d.h. weniger als $80 Daten-Bytes formatieren. Die betroffenen Sektoren sind zwar zerstört, zeigen jedoch durch ihren Status 247 ihre Existenz. Dieses Vorgehen schafft auf der Spur Platz für mehrere ganze doppelte Sektoren.

Bevor Sie einen Schutz abfragen, sollten Sie eine eventuell vorhandene Happy ausschalten. Dies beschleunigt die Abfrage von Error-Sektoren; außerdem würde eine Happy Zeitmeßergebnisse verfälschen. Danach kann sie wieder angestellt werden, um schnelleres Laden zu ermöglichen. Wie die Happy aus- und anzuschalten ist, können Sie dem Programm entnehmen.

Für die Schutzabfrage sind zwei Maschinensprache-Unterprogramme vorgesehen.

ZT = USR (ADR(ZEIT$), SEK1, PUF1, SEK2, PUF2)

Es werden zwei Sektoren in zwei verschiedene Puffer gelesen. Zurückgeliefert wird ZT. Ist ZT = 0, so ist einer der zwei Sektoren zerstört. Ansonsten bezeichnet ZT die Zeit (in 20 ms), die zum Lesen der zwei Sektoren benötigt wurde. Dieses Unterprogramm wird für die Abfrage von doppelten Sektoren und der Sektoranordnung benutzt.

STA = USR (ADR (RW$), ASC ("R"), SEK, PUF)

Es wird ein Sektor in einen Puffer gelesen. Hat STA den Wert 0, so ist der Sektor vollständig. Bei STA<>0 hat der Error-Sektor diesen Status.

Stefan Wachter

Wenn Sie die Ordnung der Sektoren auf der Diskette ein wenig manipulieren, entsteht der Schutz, der Ihre Programme vor Kopierern schützt.

Listing 1

```
10 DIM K$(1),A$(15),SIO$(11),ZEIT$(74),RW$(53),D$(1),FPRG$(855)
20 RW=ADR(RW$):ZEIT=ADR(ZEIT$):FPRG=ADR(FPRG$)
30 PF1=1024:PF2=1152:DAT=1536:PM=1664
40 EXEC INITSIO:EXEC INITUSR
50 POKE 729,10:POKE 730,2
60 ------------------------------------
70 CLS :POSITION 6,3
80 ? "Formatierer fuer Happy 1050":?
90 POKE 85,3:? "Programmiert fuers  Atari magazin":?
100 POKE 85,10:? "Von Stefan Wachter":?
110 POKE 82,8:?
120 ? "1) Diskmenue"
130 ? "2) Format-Editor"
140 ? "3) Formatieren"
150 ? "4) Sektoren testen"
160 ? "5) Floppy programmieren"
170 ? "6) Standard-Formate"
180 ? "7) Schutzabfrage testen"
190 ? "8) Happy an/aus"
200 POKE 82,6:? :? "Ihre Wahl ?";
210 POKE 82,0:EXEC GETKEY
220 IF KEY<49 OR KEY>56 THEN 210
230 CLS :?
240 ON KEY-48 EXEC DISK,FORMED,FORMAT,SEKTEST,PROGDRV,STAFMT,SCHTST,HAPPY
250 GOTO 70
260 ------------------------------------
270 REM Diskmenue
280 ------------------------------------
290 PROC DISK
300   POKE 82,5:CLS :POSITION 12,4
310   ? "Diskmenue":?
320   # WAHL
330   ? "1) Inhaltsverzeichnis"
340   ? "2) Formatierung laden"
350   ? "3) Formatierung speichern":?
360   ? "Ihre Wahl ?";
370   EXEC GETKEY
380   IF KEY<49 OR KEY>51 THEN 590
390   ? K$:? :TRAP #ERROR
400   ON KEY-48 GO# INHALT,LADEN,SPEICH
410   ------------------------------------
420   # INHALT
430   ? "Laufwerk ?";:EXEC GETKEY
440   IF KEY<49 OR KEY>50 THEN 470
450   A$="D:*.*":IF KEY=50 THEN A$="D2:*.*"
460   ? :? :DIR A$
470   ? :GO# WAHL
480   # LADEN
490   ? "Lade Datei ";:INPUT A$
500   OPEN #1,4,0,A$:BGET #1,DAT,125
510   CLOSE #1:? :GO# WAHL
520   # SPEICH
530   ? "Speichere Datei ";:INPUT A$
540   OPEN #1,8,0,A$:BPUT #1,DAT,125
550   CLOSE #1:? :GO# WAHL
560   # ERROR
570   ? "I/O Error ";ERR:? :CLOSE #1
580   GO# WAHL
590   TRAP 40000
600 ENDPROC
610 ------------------------------------
620 REM Format-Editor
630 ------------------------------------
640 PROC FORMED
650   POKE 752,1:EXEC DISP:EXEC ONPM
660   X=0:Y=0:OF=0:YA=-1
670   # LOOP
680   EXEC CLCXY:EXEC SETCUR
690   EXEC GETKEY
700   IF K$="-" THEN GO# CURUP
710   IF K$="=" THEN GO# CURDO
720   IF K$="+" THEN GO# CURLE
730   IF K$="*" THEN GO# CURRI
740   IF K$="Q" THEN GO# QUIT
750   IF KEY<48 THEN 690
760   IF KEY<58 THEN A=KEY-48:GOTO 790
770   IF KEY<65 OR KEY>70 THEN 690
780   A=KEY-55
790   POSITION CX,CY:? K$
800   B=PEEK(DAT+X+Y*30)
810   IF OF=1 THEN B=B&$F0:GOTO 830
820   B=B&$0F:A=A*16
830   B=B!A:POKE (DAT+X+Y*30),B:PAUSE 3
840   IF OF=1 THEN OF=0:GO# CURDO
850   OF=1:GO# LOOP
860   # CURUP
870   X=X-1:IF X<0 THEN X=29
880   GO# LOOP
890   # CURDO
900   X=X+1:IF X>29 THEN X=0
910   GO# LOOP
920   # CURLE
930   IF OF=1 THEN OF=0:GO# LOOP
940   OF=1:Y=Y-1:IF Y<0 THEN Y=3
950   GO# LOOP
960   # CURRI
970   IF OF=0 THEN OF=1:GO# LOOP
980   OF=0:Y=Y+1:IF Y>3 THEN Y=0
990   GO# LOOP
1000   # QUIT
1010   EXEC OUTPM:POKE 752,0
1020 ENDPROC
1030 ------------------------------------
1040 REM FORMATIEREN
1050 ------------------------------------
1060 PROC FORMAT
1070   FOR ANZ=0 TO 29
1080     IF PEEK(DAT+ANZ)=0 THEN 1100
1090   NEXT ANZ
1100   IF ANZ<>0 THEN 1120
1110   ? "Kein Sektor enthalten.":GET KEY:GOTO 1410
1120   ? ANZ;" Sektoren enthalten":? :?
1130   RESTORE #NAMEN
1140   FOR I=0 TO 3
1150     ? " ";PEEK(DAT+120+I);:POKE 85,8
1160     ? "Gap ";I+1;" (";
1170     READ A$:? A$;") ";
1180     TRAP 1220:POKE 85,29
1190     INPUT W:W=INT(W)
1200     IF W<0 OR W>255 THEN 1220
1210     POKE DAT+120+I,W
1220   NEXT I
1230   TRAP 40000:? :?
1240   D$=CHR$(PEEK(DAT+124))
1250   ? " ";D$;"  Schreibdichte ?";
1260   EXEC GETKEY
1270   IF K$="S" OR K$="E" THEN D$=K$
1280   ? D$:? :POKE DAT+124,ASC(D$)
1290   EXEC LADEDRV
1300   ? "Von Spur ";:INPUT V
1310   IF V<0 OR V>39 THEN 1410
1320   ? "Bis Spur ";:INPUT B
1330   IF B<V OR B>39 THEN 1410
1340   ? "Ok ?";:EXEC GETKEY
1350   IF K$<>"J" THEN 1410
1360   KMD=ASC("F"):STA=0:? "J":?
1370   FOR SEK=V TO B
1380     EXEC DOSIO
1390     IF X<>1 THEN ? "Formatier-Fehler":GET KEY:GOTO 1410
1400   NEXT SEK
1410 ENDPROC
1420 ------------------------------------
1430 REM SEKTOREN TESTEN
1440 ------------------------------------
1450 PROC SEKTEST
1460   KMD=ASC("S"):STA=64:PUF=PF1
1470   LAE=4:EXEC DOSIO
1480   SPT=18:A=PEEK(PF1):A=A&$A0
1490   IF A=128 THEN SPT=26:? "Enhanced Format"
1500   IF A=0 THEN ? "Single Format"
1510   ? :? "1) Teste Sektoren (von, bis)"
1520   ? "2) Teste eine Spur":?
1530   ? "Ihre Wahl ?";
1540   EXEC GETKEY
1550   IF KEY<49 OR KEY>50 THEN 1870
1560   ? K$:?
1570   IF KEY=50 THEN 1630
1580   ? "Von Sektor ";:INPUT V
1590   IF V<1 OR V>40*SPT THEN 1870
1600   ? "Bis Sektor ";:INPUT B
1610   IF B<V OR B>40*SPT THEN 1870
1620   GOTO 1660
1630   ? "Spur ";:INPUT SPUR
1640   IF SPUR<0 OR SPUR>39 THEN 1870
1650   V=SPUR*SPT+1:B=V+SPT-1
1660   ? :? "1) Error-Test"
1670   ? "2) Doppel-Test":?
1680   ? "Ihre Wahl ?";
1690   EXEC GETKEY
1700   IF KEY<49 OR KEY>50 THEN 1870
1710   ? K$:? "Mit  ESC  abbrechen":?
1720   FOR SEK=V TO B
1730     ? "Sektor ";SEK;
1740     IF KEY=50 THEN 1780
1750     X=USR(RW,ASC("R"),SEK,PF1)
1760     IF X=0 THEN ? " Ok":GOTO 1840
1770     ? " Status ";X:GOTO 1840
1780     X=USR(ZEIT,SEK,PF1,SEK,PF2)
1790     IF X=0 THEN ? " zerstoert":GOTO 1840
1800     ? "  Zeit ";X;
1810     FOR I=0 TO 126 STEP 2
1820       IF DPEEK(PF1+I)<>DPEEK(PF2+I) THEN ? "  Daten verschieden":GOTO 1840
1830     NEXT I:?
1840     IF PEEK(764)=28 THEN 1870
1850   NEXT SEK
1860   ? :GET KEY
1870 ENDPROC
1880 ------------------------------------
1890 REM Standard-Formate
1900 ------------------------------------
1910 PROC STAFMT
1920   ? "1) Single   Density"
1930   ? "2) Enhanced Density"
1940   ? "3) Beispielformat":?
1950   ? "Welches Format ?";
1960   EXEC GETKEY
1970   IF KEY<49 OR KEY>51 THEN 2290
1980   IF KEY=49 THEN SPT=18:D$="S"
1990   IF KEY=50 THEN SPT=26:D$="E"
2000   IF KEY=51 THEN 2140
2010   FOR I=0 TO 119:POKE DAT+I,0:NEXT I:S=1
2020   FOR I=0 TO SPT-1
2030     POKE DAT+I,S
2040     POKE DAT+60+I,128
2050     S=S+2
2060     IF S>SPT THEN S=2
2070   NEXT I
2080   POKE DAT+124,ASC(D$)
2090   RESTORE #GAPS
2100   FOR I=0 TO 3
2110     READ A:IF D$="E" THEN A=A*2
2120     POKE DAT+120+I,A
2130   NEXT I:GOTO 2290
2140   FOR I=0 TO 119:POKE DAT+I,0:NEXT I
2150   RESTORE #BEIFMT
2160   FOR I=0 TO 27:READ A
2170     POKE DAT+I,A:B=0
2180     IF A=4 THEN B=1
2190     IF A=5 THEN B=2
2200     IF A=6 THEN B=5
2210     POKE DAT+30+I,B
2220     POKE DAT+60+I,128:B=0
2230     IF (A=10 OR A=11) AND I>15 THEN B=16
2240     POKE DAT+90+I,B
2250   NEXT I
2260   POKE DAT+120,1:POKE DAT+121,1
2270   POKE DAT+122,40:POKE DAT+123,1
2280   POKE DAT+124,ASC("E")
2290 ENDPROC
2310 # GAPS
2320 DATA 16,6,17,14
2330 # BEIFMT
2340 DATA 1,2,3,4,5,6,7,8,9,10,11,12,13,21,15,16,17,18,19,20,14,22,23,10,11,24,25,26
2350 ------------------------------------
2360 REM Schutzabfrage testen
2370 ------------------------------------
2380 PROC SCHTST
2390   ? "1) Beispiel Abfrage"
2400   ? "2) Eigene   Abfrage":?
2410   ? "Ihre Wahl ?";
2420   EXEC GETKEY:? :?
2430   IF KEY<49 OR KEY>50 THEN 2460
2440   ON KEY-48 EXEC BEIABF,EIGABF
2450   GET KEY
2460 ENDPROC
2470 ------------------------------------
2480 PROC BEIABF
2490   SO=37*26
2500   ? "1. Error-Sektoren"
2510   STA=USR(RW,ASC("R"),SO+4,PF1)
2520   IF STA<>215 THEN 2660
2530   STA=USR(RW,ASC("R"),SO+5,PF1)
2540   IF STA<>223 THEN 2660
2550   STA=USR(RW,ASC("R"),SO+6,PF1)
2560   IF STA<>247 THEN 2660
2570   ? "2. Doppelte Sektoren"
2580   ZT=USR(ZEIT,SO+10,PF1,SO+10,PF2)
2590   IF PEEK(PF1)=PEEK(PF2) OR ZT=0 THEN 2660
2600   ZT=USR(ZEIT,SO+11,PF1,SO+11,PF2)
2610   IF PEEK(PF1)=PEEK(PF2) OR ZT=0 THEN 2660
2620   ? "3. Sektoren-Anordnung"
2630   ZT=USR(ZEIT,SO+20,PF1,SO+21,PF1)
2640   IF ZT>9 THEN 2660
2650   ? :? "Schutzabfrage Ok":GOTO 2670
2660   ? :? "Raubkopie entlarvt"
2670 ENDPROC
2680 ------------------------------------
2690 REM Eigene Abfrage kann hier
2700 REM programmiert werden
2710 ------------------------------------
```

```
2720 PROC EIGABF
2730   ? "Keine eigene Abfrage programmi
ert"
2740 ENDPROC
2750 ------------------------------------
2760 # NAMEN
2770 DATA POST INDEX
2780 DATA PRE ADDRESS
2790 DATA POST ADDRESS
2800 DATA POST DATA
2810 ------------------------------------
2820 PROC DISP
2830   FOR X=0 TO 29
2840     FOR Y=0 TO 3
2850       B=PEEK(DAT+X+Y*30)
2860       CX=2+3*Y+13*INT(X/10)
2870       CY=3+X-10*INT(X/10)
2880       EXEC HEXOUT
2890     NEXT Y
2900   NEXT X
2910   POSITION 11,18:? "Benuetze  - = +
 * Q"
2920 ENDPROC
2930 ------------------------------------
2940 PROC HEXOUT
2950   POSITION CX,CY
2960   ? HEX$(B)
2970 ENDPROC
2980 ------------------------------------
2990 PROC ONPM
3000   POKE PM,0:MOVE PM,PM+1,127:CA=0
3010   POKE 623,1:POKE 54279,4
3020   POKE 559,42:POKE 53277,2
3030   POKE 705,10
3040 ENDPROC
3050 ------------------------------------
3060 PROC OUTPM
3070   POKE 53249,0:POKE 559,34
3080   POKE 53277,0
3090 ENDPROC
3100 ------------------------------------
3110 PROC SETCUR
3120   POKE PM+CA,0:POKE 53249,CX*4+48
3130   CA=CY*4+19:POKE PM+CA,240
3140   IF Y=YA THEN 3200
3150   POSITION 14,1:YA=Y
3160   IF Y=0 THEN ? "Sektor-Nummer"
3170   IF Y=1 THEN ? "Fehler-Nummer"
3180   IF Y=2 THEN ? "Sektor-Laenge"
3190   IF Y=3 THEN ? "Sektor-Inhalt"
3200 ENDPROC
3210 ------------------------------------
3220 PROC CLCXY
3230   CX=2+OF+3*Y+13*INT(X/10)
3240   CY=3+X-10*INT(X/10)
3250 ENDPROC
3260 ------------------------------------
3270 PROC GETKEY
3280   GET KEY:KEY=KEY&127
3290   IF KEY>96 AND KEY<123 THEN KEY=KE
Y-32
3300   K$=CHR$(KEY)
3310 ENDPROC
3320 ------------------------------------
3330 REM Laufwerk programmieren
3340 ------------------------------------
3350 PROC PROGDRV
3360   X=USR(RW,ASC("R"),$9780,PF1)
3370   FOR I=0 TO $1F
3380     IF PEEK(PF1+$20+I)=ASC("F") THE
N 3520
3390     IF PEEK(PF1+$20+I)=0 THEN 3420
3400   NEXT I
3410   ? "PRG-ERR":STOP
3420   POKE PF1+$20+I,ASC("F")
3430   POKE PF1+$40+I,0
3440   POKE PF1+$60+I,$90
3450   X=USR(RW,ASC("W"),$9780,PF1)
3460   IF PEEK(FPRG)=0 THEN EXEC PRGSTR
3470   FOR I=0 TO 7
3480     X=USR(RW,ASC("W"),$9000+I*$80,F
PRG+I*$80)
3490   NEXT I
3500   KMD=ASC("H"):STA=0:SEK=$6060
3510   EXEC DOSIO
3520 ENDPROC
3530 ------------------------------------
3540 REM Happy an- und ausschalten
3550 ------------------------------------
3560 PROC HAPPY
3570   ? "1) Happy anschalten"
3580   ? "2) Happy ausschalten":?
3590   ? "Ihre Wahl ?";
3600   EXEC GETKEY
3610   IF KEY<49 OR KEY>50 THEN 3650
3620   KMD=ASC("H"):STA=0
3630   SEK=3:IF KEY=50 THEN SEK=$6060
3640   EXEC DOSIO
3650 ENDPROC
3660 ------------------------------------
3670 PROC LADEDRV
3680   POKE DAT+125,ANZ
3690   X=USR(RW,ASC("W"),$8000,DAT)
3700 ENDPROC
3710 ------------------------------------
3720 PROC DOSIO
3730   POKE $0300,49:POKE $0301,1
3740   POKE $0302,KMD:POKE $0303,STA
3750   DPOKE $0304,PUF:DPOKE $0306,2
3760   DPOKE $0308,LAE:DPOKE $030A,SEK
3770   X=USR(ADR(SIO$))
3780 ENDPROC
3790 ------------------------------------
3800 PROC INITSIO
3810   RESTORE #SIODAT
3820   FOR I=0 TO 10
3830     READ A:POKE ADR(SIO$)+I,A
3840   NEXT I
3850 ENDPROC
3860 # SIODAT
3870 DATA 104,32,89,228,132,212
3880 DATA 169,0,133,213,96
3890 ------------------------------------
3900 PROC INITUSR
3910   RESTORE #USRDAT
3920   FOR I=0 TO 73
3930     READ A
3940     POKE ZEIT+I,A
3950   NEXT I
3960   FOR I=0 TO 52
3970     READ A
3980     POKE RW+I,A
3990   NEXT I
4000 ENDPROC
4010 # USRDAT
4020 DATA 104,169,0,133,212,133,213,104
4030 DATA 141,11,3,104,141,10,3,104
4040 DATA 141,5,3,104,141,4,3,169
4050 DATA 82,141,2,3,169,1,141,1
4060 DATA 3,32,83,228,16,5,104,104
4070 DATA 104,104,96,169,0,133,20,104
4080 DATA 141,11,3,104,141,10,3,104
4090 DATA 141,5,3,104,141,4,3,32
4100 DATA 83,228,16,1,96,165,20,133
4110 DATA 212,96
4120 DATA 104,169,0,133,212,133,213,104
4130 DATA 104,141,2,3,104,141,11,3
4140 DATA 104,141,10,3,104,141,5,3
4150 DATA 104,141,4,3,169,1,141,1
4160 DATA 3,32,83,228,48,1,96,169
4170 DATA 83,141,2,3,32,83,228,173
4180 DATA 235,2,133,212,96
4190 ------------------------------------
4200 PROC PRGSTR
4210   ? "Lese Daten ein":?
4220   RESTORE #FLPDAT
4230   FOR I=0 TO 60:SUM=0
4240     FOR J=0 TO 13
4250       READ A:POKE FPRG+I*14+J,A:SUM
=SUM+A
4260     NEXT J
4270     READ A:IF SUM<>A THEN ? "Datenf
ehler in ";I;". Datenzeile.":? "Summe wa
r ";SUM;" anstatt ";A:STOP
4280   NEXT I
4290   POKE FPRG+854,0
4300 ENDPROC
4310 # FLPDAT
4320 DATA 173,124,128,201,83,208,7,173,1
28,2,9,32,208,5,1481
4330 DATA 173,128,2,41,223,141,128,2,160
,29,185,90,128,73,1503
4340 DATA 255,153,90,128,136,16,245,32,9
8,243,44,0,4,48,1492
4350 DATA 34,112,32,32,57,242,165,130,13
3,141,32,236,242,169,1757
4360 DATA 15,141,159,2,169,240,141,0,4,1
60,0,173,124,128,1456
4370 DATA 201,83,240,6,76,135,145,76,148
,244,44,128,2,16,1544
4380 DATA 251,169,0,141,3,4,44,128,2,16,
251,141,3,4,1157
4390 DATA 169,210,141,159,2,162,172,169,
0,44,128,2,112,3,1473
4400 DATA 76,148,244,16,246,141,3,4,202,
208,240,169,252,44,1993
4410 DATA 128,2,112,3,76,148,244,16,246,
141,3,4,174,120,1417
4420 DATA 128,169,0,44,128,2,112,3,76,14
8,244,16,246,141,1457
4430 DATA 3,4,202,208,240,174,121,128,16
9,0,44,128,2,112,1535
4440 DATA 3,76,148,244,16,246,141,3,4,20
2,208,240,169,254,1954
4450 DATA 44,128,2,112,3,76,148,244,16,2
46,141,3,4,165,1332
4460 DATA 141,44,128,2,112,3,76,148,244,
16,246,141,3,4,1308
4470 DATA 169,0,44,128,2,112,3,76,148,24
4,16,246,141,3,1332
4480 DATA 4,185,0,128,44,128,2,112,3,76,
148,244,16,246,1336
4490 DATA 141,3,4,169,0,44,128,2,112,3,7
6,148,244,16,1090
4500 DATA 246,141,3,4,190,30,128,189,75,
147,44,128,2,112,1439
4510 DATA 3,76,148,244,16,246,141,3,4,17
4,122,128,169,0,1474
4520 DATA 44,128,2,112,3,76,148,244,16,2
46,141,3,4,202,1369
4530 DATA 208,240,190,30,128,189,69,147,
44,128,2,112,3,76,1566
4540 DATA 148,244,16,246,141,3,4,190,60,
128,185,90,128,44,1627
4550 DATA 128,2,112,3,76,148,244,16,246,
141,3,4,202,208,1533
4560 DATA 240,190,30,128,189,81,147,44,1
28,2,112,3,76,148,1518
4570 DATA 244,16,246,141,3,4,174,123,128
,169,0,44,128,2,1422
4580 DATA 112,3,76,148,244,16,246,141,3,
4,202,208,240,200,1843
4590 DATA 204,125,128,176,3,76,159,144,1
62,0,76,31,147,44,1475
4600 DATA 128,2,16,251,169,78,141,3,4,44
,128,2,16,251,1233
4610 DATA 141,3,4,169,210,141,159,2,162,
144,169,78,44,128,1554
4620 DATA 2,112,3,76,148,244,16,246,141,
3,4,202,208,240,1645
4630 DATA 162,0,169,78,44,128,2,112,3,76
,148,244,16,246,1428
4640 DATA 141,3,4,202,208,240,162,12,169
,0,44,128,2,112,1427
4650 DATA 3,76,148,244,16,246,141,3,4,20
2,208,240,162,3,1696
4660 DATA 169,246,44,128,2,112,3,76,148,
244,16,246,141,3,1578
4670 DATA 4,202,208,240,169,252,44,128,2
,112,3,76,148,244,1832
4680 DATA 16,246,141,3,4,174,120,128,169
,78,44,128,2,112,1365
4690 DATA 3,76,148,244,16,246,141,3,4,20
2,208,240,174,121,1826
4700 DATA 128,169,0,44,128,2,112,3,76,14
8,244,16,246,141,1457
4710 DATA 3,4,202,208,240,162,3,169,245,
44,128,2,112,3,1525
4720 DATA 76,148,244,16,246,141,3,4,202,
208,240,169,254,44,1995
4730 DATA 128,2,112,3,76,148,244,16,246,
141,3,4,165,141,1429
4740 DATA 44,128,2,112,3,76,148,244,16,2
46,141,3,4,169,1336
4750 DATA 0,44,128,2,112,3,76,148,244,16
,246,141,3,4,1167
4760 DATA 185,0,128,44,128,2,112,3,76,14
8,244,16,246,141,1473
4770 DATA 3,4,169,0,44,128,2,112,3,76,14
8,244,16,246,1195
4780 DATA 141,3,4,190,30,128,189,75,147,
44,128,2,112,3,1196
4790 DATA 76,148,244,16,246,141,3,4,174,
122,128,169,78,44,1593
4800 DATA 128,2,112,3,76,148,244,16,246,
141,3,4,202,208,1533
4810 DATA 240,162,3,169,245,44,128,2,112
,3,76,148,244,16,1592
4820 DATA 246,141,3,4,202,208,240,190,30
,128,189,69,147,44,1841
4830 DATA 128,2,112,3,76,148,244,16,246,
141,3,4,190,60,1373
4840 DATA 128,185,90,128,44,128,2,112,3,
76,148,244,16,246,1550
4850 DATA 141,3,4,202,208,240,190,30,128
,189,81,147,44,128,1735
4860 DATA 2,112,3,76,148,244,16,246,141,
3,4,174,123,128,1420
4870 DATA 169,78,44,128,2,112,3,76,148,2
44,16,246,141,3,1410
4880 DATA 4,202,208,240,200,204,125,128,
176,3,76,18,146,162,1892
4890 DATA 78,169,1,45,0,4,240,10,44,128,
2,16,246,142,1125
4900 DATA 3,4,48,241,173,150,2,173,0,4,7
2,32,98,243,1243
4910 DATA 104,41,4,208,3,76,143,244,76,1
48,244,251,248,248,2038
4920 DATA 251,0,251,247,247,247,0,247,24
7,247,0,247,247,247,2725
```

Listing 2

```
        .OPT NO LIST
;
; Neuer Befehl F
; Formatiert eine Spur mit selbst
; definiertem Format
;
DAUX      =   $82
SPUR      =   $8D
;
PORTA     =   $0280
BEFSTA    =   $0400
DATEN     =   $0403
;
CONRES    =   $F362
MOTAN     =   $F239
KOPFSPUR  =   $F2EC
SENDCPL   =   $F48F
SENDERR   =   $F494
;
PUF       =   $8000    ; Format-Daten
STAB      =   PUF      ; Sektor-Nummer
ETAB      =   PUF+30   ; Fehler-Nummer
LTAB      =   PUF+60   ; Sektor-Laenge
FTAB      =   PUF+90   ; Sektor-Inhalt
GP1       =   PUF+120  ; POST INDEX
GP2       =   PUF+121  ; PRE ADDRESS
GP3       =   PUF+122  ; POST ADDRESS
GP4       =   PUF+123  ; POST DATA
DEN       =   PUF+124  ; Sng oder Enh
ANZ       =   PUF+125  ; Anzahl der Sek.
;
; Schreibe Akku auf Diskette
;
        .MACRO WRT1
?W1     BIT PORTA
        BVS *+5        ; Fehler ?
        JMP SENDERR
        BPL ?W1        ; Data Request ?
        STA DATEN
        .ENDM
;
; Schreibe X-mal Akku auf Diskette
;
        .MACRO WRT
?W2     BIT PORTA
        BVS *+5        ; Fehler ?
        JMP SENDERR
        BPL ?W2        ; Data Request ?
        STA DATEN
        DEX
        BNE ?W2


        .ENDM
;
        *=  $9000
;
        LDA DEN
        CMP #'S
        BNE ED
        LDA PORTA      ; Aktiviere FM
        ORA #$20       ; Modus
        BNE ?LO
ED      LDA PORTA      ; Aktiviere MFM
        AND #$DF       ; Modus
?LO     STA PORTA
;
        LDY #29        ; Der Sektorinhalt
?L1     LDA FTAB,Y     ; muss wegen dem
        EOR #$FF       ; "inverted Data
        STA FTAB,Y     ; Bus" noch
        DEY            ; "inverted"
        BPL ?L1        ; werden
;
        JSR CONRES     ; FDC reset
        BIT BEFSTA
        BMI ERR        ; Write Protect
        BVS ERR        ; Not ready
        JSR MOTAN      ; Motor anschalten
        LDA DAUX       ; Schreib-/Lese-
        STA SPUR       ; Kopf auf Spur
        JSR KOPFSPUR   ; ruecken
        LDA #15        ; Timer fuer 1.
        STA $029F      ; Indeximp. setzen
        LDA #$F0       ; WRITE-TRACK
        STA BEFSTA     ; Kommando an FDC
;
        LDY #0         ; Pointer auf 0
        LDA DEN
        CMP #'S
        BEQ SNG
        JMP ENH
;
ERR     JMP SENDERR
;
SNG     BIT PORTA      ; Formatiere in
        BPL SNG        ; Single Density
        LDA #0
        STA DATEN
?S0     BIT PORTA
        BPL ?S0
        STA DATEN
        LDA #$D2       ; Timer auf 210
        STA $029F      ; ms setzen
;
        LDX #$AC       ; $ac Gapbytes
        LDA #0
         WRT
        LDA #$FC       ; INDEX MARK
         WRT1
        LDX GP1        ; POST INDEX
        LDA #0
         WRT
;
?S1     LDX GP2        ; PRE ADDRESS
        LDA #0
         WRT
;
; Formatiere einen Header
;
        LDA #$FE       ; ADDRESS MARK
         WRT1
        LDA SPUR       ; Spur-Nummer
         WRT1
        LDA #0         ; Seiten-Nummer
         WRT1
        LDA STAB,Y     ; Sektor-Nummer
         WRT1
        LDA #0         ; Sektor-Laenge
         WRT1
        LDX ETAB,Y     ; Die Pruefsummen-
        LDA ETB2,X     ; bytes dienen zur
         WRT1          ; Fehlererzeugung
;
        LDX GP3        ; POST ADDRESS
        LDA #0
         WRT
;
; Formatiere ein Datenfeld
;
        LDX ETAB,Y     ; DATA ADDRESS
        LDA ETB1,X     ; dient zur
         WRT1          ; Fehlererzeugung
        LDX LTAB,Y     ; Sektor-Laenge
        LDA FTAB,Y     ; Sektor-Inhalt
         WRT
        LDX ETAB,Y     ; Die Pruefsummen-
        LDA ETB3,X     ; bytes dienen zur
         WRT1          ; Fehlererzeugung
        LDX GP4        ; POST DATA
        LDA #0
         WRT
        INY            ; Naechster Sektor
        CPY ANZ        ; Fertig ?
        BCS *+5
        JMP ?S1
        LDX #0
        JMP FEN
;
ENH     BIT PORTA      ; Formatiere in
        BPL ENH        ; Enhanced Density
        LDA #$4E
        STA DATEN
?E0     BIT PORTA
        BPL ?E0
        STA DATEN
        LDA #$D2       ; Timer auf 210 ms
        STA $029F      ; setzen
;
        LDX #$90       ; $190 Gapbytes
        LDA #$4E
         WRT
        LDX #0
        LDA #$4E
         WRT
        LDX #$0C       ; $c   Gapbytes
        LDA #0
         WRT
        LDX #3         ; $c2 ohne Takt
        LDA #$F6
         WRT
        LDA #$FC       ; INDEX MARK
         WRT1
        LDX GP1        ; POST INDEX
        LDA #$4E
         WRT
;
?E1     LDX GP2        ; PRE ADDRESS
        LDA #0
         WRT
;
; Formatiere einen Header
;
        LDX #3         ; $a1 ohne Takt
        LDA #$F5
         WRT
        LDA #$FE       ; ADDRESS MARK
         WRT1
        LDA SPUR       ; Spur-Nummer
         WRT1
        LDA #0         ; Seiten-Nummer
         WRT1
        LDA STAB,Y     ; Sektor-Nummer
         WRT1
        LDA #0         ; Sektor-Laenge
         WRT1
        LDX ETAB,Y     ; Die Pruefsummen-
        LDA ETB2,X     ; bytes dienen zur
         WRT1          ; Fehlererzeugung
;
        LDX GP3        ; POST ADDRESS
        LDA #$4E
         WRT
;
; Formatiere ein Datenfeld
;
        LDX #3         ; $a1 ohne Takt
        LDA #$F5
         WRT
        LDX ETAB,Y     ; DATA ADDRESS
        LDA ETB1,X     ; dient zur
         WRT1          ; Fehlererzeugung
        LDX LTAB,Y     ; Sektor-Laenge
        LDA FTAB,Y     ; Sektor-Inhalt
         WRT
        LDX ETAB,Y     ; Die Pruefsummen-
        LDA ETB3,X     ; bytes dienen zur
         WRT1          ; Fehlererzeugung
        LDX GP4        ; POST DATA
        LDA #$4E
         WRT
        INY            ; Naechster Sektor
        CPY ANZ        ; Fertig ?
        BCS *+5
        JMP ?E1
        LDX #$4E
;
FEN     LDA #1
?F1     AND BEFSTA     ; FDC busy ?
        BEQ ?F2        ; nicht mehr busy
        BIT PORTA      ; Data Request ?
        BPL ?F1        ; Nein
        STX DATEN
        BMI ?F1        ; Immer
?F2     LDA $0296
        LDA BEFSTA
        PHA            ; Save FDC-Status
        JSR CONRES     ; FDC reset
        PLA
        AND #4         ; Datenverlust ?
        BNE *+5        ; Ja
        JMP SENDCPL
        JMP SENDERR
;
ETB1    .BYTE $FB,$FB,$FB,$FB,$00,$FB
ETB2    .BYTE $F7,$F7,$F7,$00,$F7,$F7
ETB3    .BYTE $F7,$00,$F7,$F7,$F7,$00
```

Listing 3

```
        .OPT NO LIST
;
; Unterprogramm ZEIT
; Zum Abfragen von doppelten Sektor:
; und der Sektor-Anordnung
;
        *=  $0600
        PLA
        LDA #0
        STA $D4
        STA $D5
        PLA          ; Sektor 1
        STA $030B
        PLA
        STA $030A
        PLA          ; Puffer 1
        STA $0305
        PLA
        STA $0304
        LDA #'R
        STA $0302
        LDA #1
        STA $0301
        JSR $E453
        BPL OK
        PLA
        PLA
        PLA
        PLA
        RTS
OK      LDA #0
        STA 20
        PLA          ; Sektor 2
        STA $030B
        PLA
        STA $030A
        PLA          ; Puffer 2
        STA $0305
        PLA
        STA $0304
        JSR $E453
        BPL *+3
        RTS
        LDA 20
        STA $D4
        RTS
```

Listing 4

```
        .OPT NO LIST
;
; Unterprogramm RW
; Zum Abfragen von Errorsektoren
        *=  $0600
        PLA
        LDA #0
        STA $D4
        STA $D5
        PLA          ; Kommando
        PLA
        STA $0302
        PLA          ; Sektor
        STA $030B
        PLA
        STA $030A
        PLA          ; Pufferadresse
        STA $0305
        PLA
        STA $0304
        LDA #1
        STA $0301
        JSR $E453
        BMI *+3
        RTS
        LDA #'S
        STA $0302
        JSR $E453
        LDA 747
        STA $D4
        RTS
```

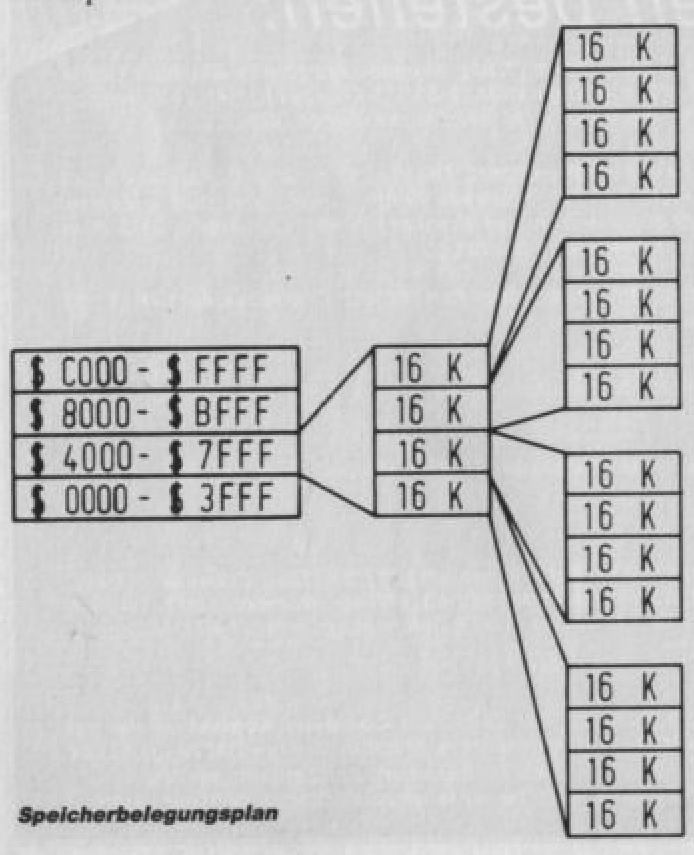
320 KByte im Eigenbau

Welcher 800-XL-Besitzer war nicht schon einmal neidisch auf die Vorzüge des 130 XE? Mit dessen zusätzlichem Speicher werden Datenzugriffe und -transfer enorm erleichtert und beschleunigt! Es sind nun zwar Aufrüstplatinen auf bis zu 320 KByte zu erwerben, jedoch dürfte ihr Preis mit ca. 150 bis 250 DM für viele zu hoch sein.

8 Bit

Wer aber einen Lötkolben bedienen kann und etwas Zeit und ca. 80 DM aufwendet, kann seinen 800 XL auch im Do-it-yourself-Verfahren auf 320 KByte erweitern – und das auch noch größtenteils kompatibel zum 130 XE!

Es soll aber gleich darauf hingewiesen werden, daß der Einbau dieser Speichererweiterung wirklich gute Kenntnisse im Umgang mit dem Lötkolben voraussetzt. Ein Fehler kann zur Beschädigung bzw. Zerstörung des Computers führen. Selbstverständlich erlischt bei Eingriffen in das Gerät auch jeglicher Garantieanspruch.



Speicherbelegungsplan

Grundlagen

Jeder, der sich intensiver mit seinem Computer auseinandergesetzt hat, wird wissen, daß in den Atari-Computern die CPU 6502C verwendet wird. Dabei handelt es sich um eine 8-Bit-CPU, die 16 Adreßleitungen besitzt. Mit ihnen lassen sich 64 KByte (65536 = 2^{16} Byte) Speicher adressieren. Mehr ist hier eigentlich nicht möglich.

Der Trick besteht nun darin, in einem Speicherbereich der 64 KByte das interne RAM auszublenden und dafür das zusätzliche einzublenden. Dies wird durch Einschreiben eines bestimmten Wertes in Adresse $D301 (54017) gesteuert. Die Adresse $D301 ist bei den Atari-Computern der Port B des eingebauten 6520-Bausteins (PIA), der zwei bidirektionale, 8 Bit breite Ports besitzt.

Der Port B wird schon von Hause aus zur Speicherverwaltung benutzt; daher ist er für die Aufgabe, den neuen Speicher zu adressieren, geradezu prädestiniert.

Die Belegung der Bits bei Port B

Bit	800 XL	130 XE	320-K-Erweiterung
PB0	Betriebssystem ein- oder ausschalten		
PB1	Basic-ROM ein- oder ausschalten		
PB2 PB3	unbenutzt	Auswahl von einer von 4 möglichen 16K-RAM-Banks	wie beim 130 XE
PB4	unbenutzt	0=CPU hat Zugriff auf das zus. RAM	0=CPU und Antic haben Zugriff auf das zus. RAM
PB5	unbenutzt	0=Antic hat Zugriff auf das zus. RAM	Auswahl von einer von 4 möglichen 64K-RAM-Banks
PB6	unbenutzt	unbenutzt	
PB7	Selbsttest nach $5000 spiegeln		

Es fällt auf, daß bei der 320-KByte-Erweiterung der Antic- bzw. CPU-Zugriff nicht mehr getrennt zugelassen werden kann. Mir ist jedoch kein Programm bekannt, das diese Unterscheidungsmöglichkeit benutzt. Somit dürfte sich die Inkompatibilität auf wenige Spezialanwendungen beschränken!

PB5 und PB6 stellen stattdessen die fehlenden zwei Adreßleitungen für die zusätzlichen 256 KByte (2^{18}).

Vorsichtsmaßnahmen und Hinweise

Bevor mit dem Zusammenbau begonnen wird, hier noch einige Regeln, die zu beachten sind.

Nochmals sei darauf hingewiesen, daß mit dem Eingriff in den Rechner jeglicher Garantieanspruch erlischt. Auch ist für die Arbeit Erfahrung und Geschick im Löten erforderlich. Geeignet ist ein Lötkolben mit feiner Spitze und maximal 30 Watt, besser noch eine Lötstation mit regelbarer Temperatur.

Die RAM-Bausteine sind statischer Aufladung gegenüber empfindlich. Um ihre Zerstörung zu vermeiden, hier einige Regeln:

1. Die RAMs auf eine leitende Unterlage legen (falls nicht vorhanden, Abschirmblech des Rechners)
2. RAM-Bausteine möglichst nicht an den Beinchen anfassen
3. Rechnerplatine (Masse) leitend mit der Unterlage verbinden
4. Wenn möglich, Lötkolben (Potentialausgleichsbuchse) und eigenen Körper mit der Masse der Rechnerplatine leitend verbinden. Man sollte möglichst oft mit den Fingern einen Massepunkt an der Rechnerplatine berühren; dadurch entsteht auch ein Potentialausgleich.

Diese Vorsichtsmaßnahmen tragen dazu bei, daß man nicht nachträglich durch statische Aufladung zerstörte RAMs austauschen muß.

Hardware

Es werden nur Standardbauteile verwendet:

1×74LS32	4fach-Oder
1×74LS08	4fach-Und
1×74LS04	6fach-Inverter
1×74LS157	Multiplexer
8×41256-150ns	RAM 256 KBit (z.B. NEC D41256C-15)

Es können natürlich auch RAMs anderer Hersteller verwendet werden (z.B. Motorola, Hitachi usw.)

Hier möchte ich noch etwas zum Preis sagen. Die Bauteile sollten insgesamt nicht viel mehr als 80 DM kosten (ca. 8 DM pro Stück). Falls dies nicht möglich ist, lohnt es sich auf jeden Fall, sie bei einem Elektronikversand zu bestellen. (Anzeigen finden Sie in Elektronikzeitschriften.) Noch ein Tip: Bei Sammelbestellungen wird oft ein Rabatt gewährt.

Aufbau der Erweiterung

Zunächst fertigt man von dem abgedruckten Layout eine Platine (in spiegelbildlicher Form!). Es bietet sich die Verwendung einer Lochrasterplatine an, da hier Ätz- und Bohrarbeiten nicht mehr notwendig sind. Die ICs auf der Platine sollten wegen der Höhe (!) nicht gesockelt werden, sonst bekommt man Probleme mit dem Abschirmblech. Vom Multiplexer (74LS157) werden folgende Pins hochgebogen:

1, 2, 3, 4, 7, 9, 10, 12, 13;

Pin 7 wird möglichst dicht am IC abgekniffen!

Nun lötet man das IC "huckepack" auf U 27 (auf korrekte Richtung achten!), R 108 wird auf der Rechnerplatine dicht am Widerstand abgekniffen (er befindet sich links unterhalb der RAMs). Die verbleibenden Anschlußstücke erhalten folgende Bezeichnungen:

zur Platinenmitte hin = RA
zur Außenseite hin = RB

Dann werden die RAM-Bausteine unter Beachtung der beschriebenen Vorsichtsmaßnahmen "huckepack" auf die bereits vorhandenen 8 RAMs aufgelötet. (Auf Einkerbung oder Markierung achten!) Zuvor werden jedoch die folgenden Pins hochgebogen:

RAM-Pin 1, 9, 13, 15

Jeweils nummernmäßig gleiche Pins werden durch isolierte Litze miteinander verdrahtet.

Die verdrahteten Pins werden dann folgendermaßen angeschlossen:

Pin		Anschluß
1	an	74LS157-Pin 4
9	an	74LS157-Pin 9
13	an	74LS157-Pin 12
15	an	Platinen-Anschluß c

Verbindungen von und zur Platine:

Platinen-Anschluß	wird verbunden mit
a	RA
b	RB
c	RAM-Pin 15
d	74LS157-Pin 2
e	74LS157-Pin 3
f	74LS157-Pin 10
g	74LS157-Pin 13
h	PIA-Pin 12 (PB2)
i	PIA-Pin 13 (PB3)
k	PIA-Pin 14 (PB4)
l	PIA-Pin 15 (PB5)
m	PIA-Pin 16 (PB6)
n	CPU-Pin 24 (A14)
o	CPU-Pin 25 (A15)
+	+5 Volt
–	Masse (GND)

Die 5-Volt-Spannung kann am Elektrolytkondensator neben der Netzteilbuchse abgenommen werden. A14 und A15 werden durch Einstecken von Drähten in den Prozessorsockel angezapft.

Es empfiehlt sich, die Leitung vom PIA-PB4 zu unterbrechen und einen Schalter einzubauen. Damit kann man die RAM-Erweiterung jederzeit ausschalten; der Rechner ist dann wieder ein ganz normaler 800 XL.

Test der Erweiterung

Dieser Test sollte vor dem Zusammenbau des Rechnergehäuses erfolgen! Es empfiehlt sich, vor dem Anschalten folgende Punkte noch einmal durchzugehen:

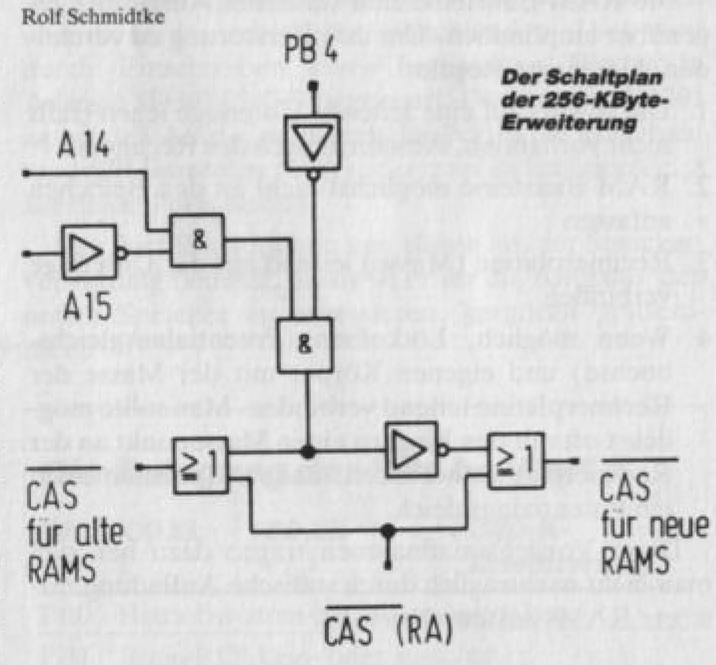
Auf Kurzschlüsse oder schlechte Lötstellen untersuchen!
Korrektes Layout? Die Platinenunterseite muß spiegelbildlich dem abgebildeten Layout entsprechen!
Richtige Verdrahtung anhand der Tabelle kontrollieren!
Bestückung der Platine kontrollieren (Alle ICs richtig herum eingesetzt?).

Wurde kein Fehler gefunden, werden das Flachbandkabel der Tastatur wieder eingesteckt und Netzteil und Fernseher (Monitor) angeschlossen.

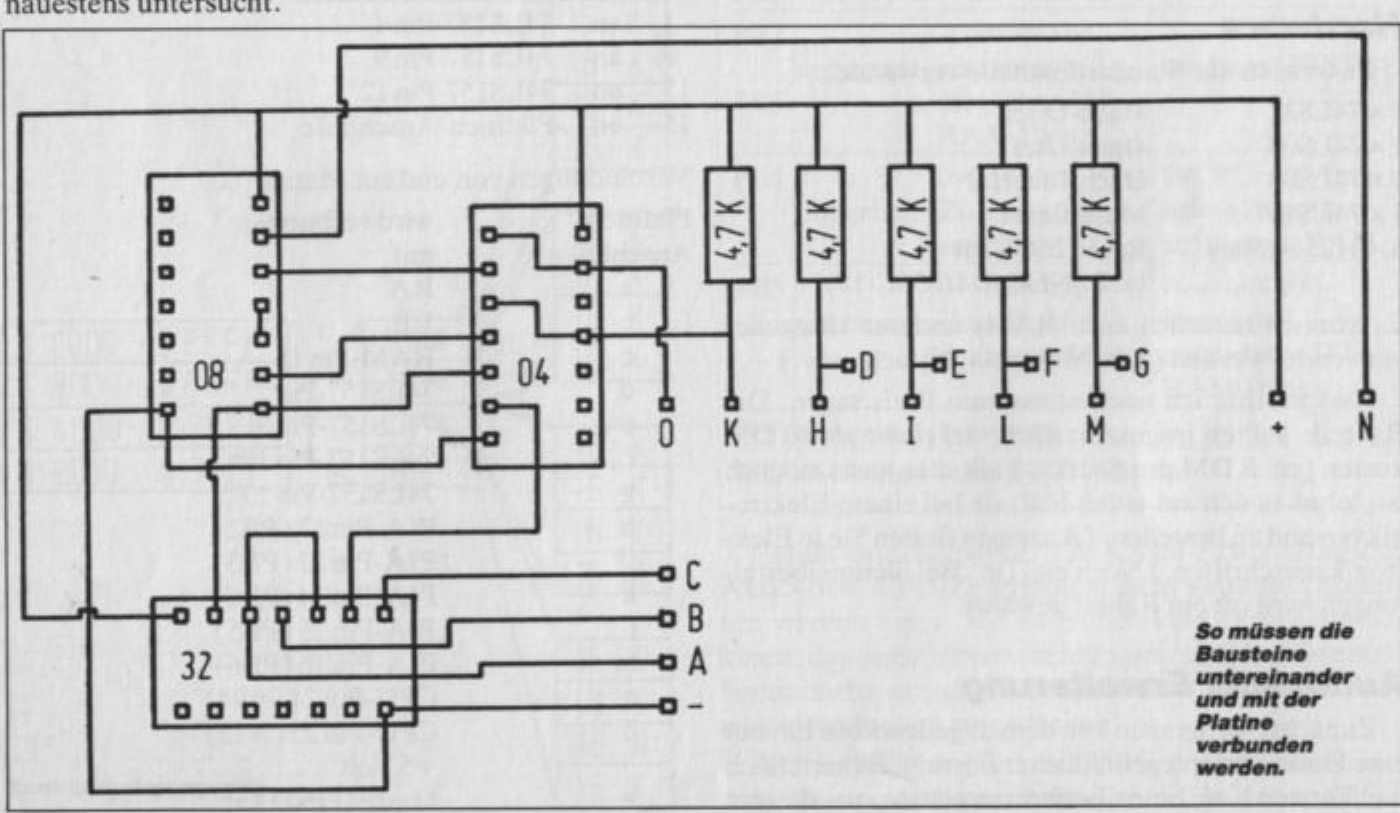
Sollte nach Einschalten des Rechners nicht der übliche GRAPHICS-0-Bildschirm erscheinen, so wird sofort (!) wieder ausgeschaltet und alles noch einmal genauestens untersucht.

Nun noch zur Funktionsprüfung der Erweiterung. Man lädt ein DOS, das die RAM-Disk des 130 XE unterstützt (z.B. DOS2.5 mit RAMDISK.COM), und überprüft durch verschiedene Zugriffe auf die RAM-Disk (D8:). Eine andere Möglichkeit besteht darin, mit einem Maschinensprachemonitor Adresse $D301 zu verändern und jeweils Schreib/Leseoperationen im Bereich von $4000-$7FFF durchzuführen. Die dritte und sicherste Möglichkeit ist, das abgedruckte Basic-Programm einzutippen und zu starten.

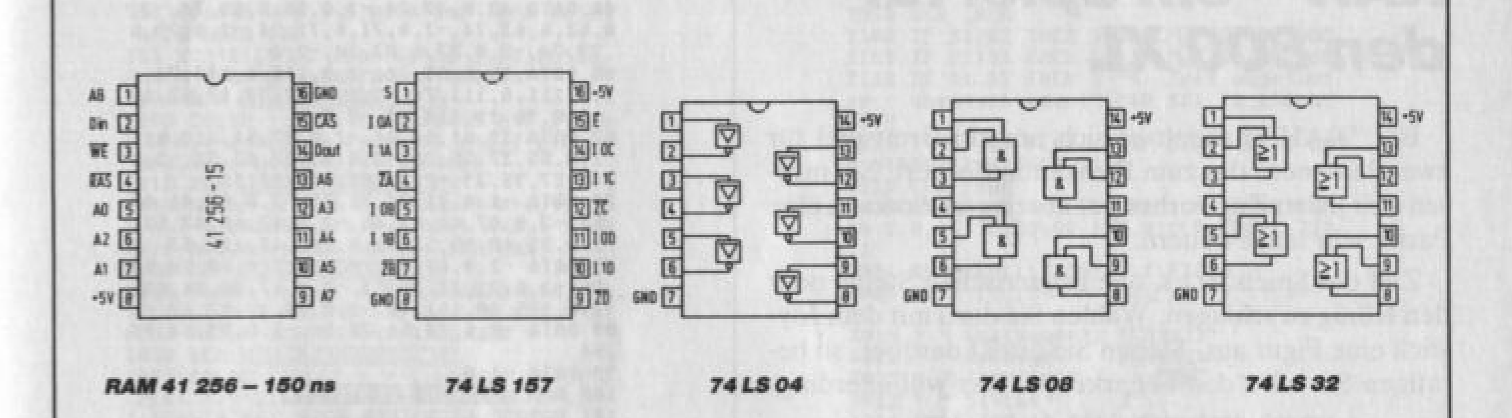
Rolf Schmidtke



Der Schaltplan der 256-KByte-Erweiterung



So müssen die Bausteine untereinander und mit der Platine verbunden werden.



RAM 41 256 – 150 ns   74 LS 157   74 LS 04   74 LS 08   74 LS 32

Testprogramm

```
1 POKE 54Ø17,253
5 DIM JN$(1)
1Ø PRINT CHR$(125)
2Ø POSITION 1Ø,5:? "256KB RAM-ERWEITERUN
GS"
3Ø POSITION 1Ø,7:? "     TEST-PROGRAMM"
4Ø POSITION 1Ø,9:? "(C)86 BY RS INDUSTRI
ES"
5Ø POSITION Ø,2Ø:? " ZUM START DES TESTS
-TASTE DRUECKEN..."
6Ø POKE 752,1:?
7Ø A=PEEK(764):IF A=255 THEN 7Ø
8Ø PIA=54Ø17
82 ? CHR$(125):POSITION 1Ø,3:? "TEST lae
uft..."
83 POKE 752,Ø
84 POKE 764,255
85 RESTORE
86 POSITION 2,5:? "ES WIRD GESCHRIEBEN..
."
87 FOR X=Ø TO 16
9Ø ? "BANK";X-1
95 READ PIAWERT:POKE PIA,PIAWERT
1ØØ FOR I=Ø TO 15:POKE 16384+I,X:NEXT I
11Ø NEXT X
12Ø RESTORE
125 POSITION 2,5:? "ES WIRD GELESEN...
"
13Ø FOR X=Ø TO 16
135 ? "ﾂﾁﾎﾋ";X-1
14Ø READ PIAWERT:POKE PIA,PIAWERT
15Ø FOR I=Ø TO 15:IF PEEK(16384+I)<>X TH
EN GOTO 2ØØ
155 NEXT I
16Ø NEXT X
17Ø POKE PIA,253
175 GOSUB 25Ø
18Ø ? CHR$(125)
19Ø POSITION 1Ø,5:? "HERZLICHEN GLUECKWU
NSCH!!!"
195 POSITION 1Ø,7:? "DIE RAM-ERWEITERUNG
"
196 POSITION 1Ø,9:? "FUNKTIONIERT EINWAN
DFREI!!"
197 POSITION 1Ø,11:? "INS DOS MENUE?":IN
PUT JN$
198 IF JN$="J" THEN DOS
199 END
2ØØ ? CHR$(125),"     FEHLERRRRRRRR...."
2Ø5 FØ=X:F1=I:CD=PEEK(16384+I)
21Ø FL=ABS(CD-FØ)
212 IF FL>16 THEN GOSUB 25Ø:GOTO 197
215 GOTO 3ØØ
25Ø ? CHR$(125);:POSITION 5,5:? "NUN NOC
H EIN TEST DER EINZELNEN RAMS..."
251 FOR ZWEI=Ø TO 255 STEP 255
252 RESTORE
255 FOR X=Ø TO 16
256 READ PIAWERT:POKE PIA,PIAWERT
26Ø FOR I=Ø TO 15:POKE 16384+I,ZWEI:NEXT
 I
28Ø NEXT X
281 RESTORE
282 FOR X=Ø TO 16
283 READ PIAWERT:POKE PIA,PIAWERT
284 FOR I=Ø TO 15:IF PEEK(16384)<>ZWEI T
HEN DEF=ABS(PEEK(16384+I)-ZWEI)
285 NEXT I
286 NEXT X
29Ø NEXT ZWEI
291 ? CHR$(125):FOR K=7 TO Ø STEP -1
292 IF DEF>=2^K THEN ? "RAM AUF 1";9+K;"
 IST DEFEKT ODER NICHT RICHTIG ANGELOETE
T !":DEF=DEF-2^K
293 NEXT K
294 RETURN
295 END
3ØØ IF FL=16 THEN 316
31Ø PRINT "FEHLER IN DER BANKAUSWAHL,KON
TROLLIERE LEITUNGEN ZUR PIA UND ZUM MULT
IPLEXER !"
312 GOTO 197
316 PRINT "DIE GESAMTE ERWEITERUNG KANN
NICHT ANGESPROCHEN WERDEN!KONTROLLIERE P
LATINE UND LEITUNGEN ZUR PIA";
317 GOTO 197
5ØØ DATA 253,129,133,137,141,161,165,169
,173,193,197,2Ø1,2Ø5,225,229,233,237
```

KAH – ein Spiel für den 800 XL

Bei "KAH" handelt es sich um ein Brettspiel für zwei Personen, das zum Denken auffordert. Sie müssen sich jeden Zug vorher gut überlegen. So kann eine Partie sehr lange dauern.

Ziel des Spiels ist es, alle gegnerischen Steine oder den König zu schlagen. Wählen Sie dazu mit dem Joystick eine Figur aus. Stehen Sie direkt darüber, so betätigen Sie kurz den Feuerknopf. Das will allerdings

8 Bit

gelernt sein, denn drücken Sie ihn zu lange, wird die Figur wieder abgewählt. Andererseits ist das aber recht praktisch, wenn man sich für eine falsche Figur entschieden hat. Vor dem Spiel können Sie unter folgenden Optionen wählen:

Könige (Ja/Nein)	Wählen Sie, ob Sie mit Königen spielen wollen. Bei Nein müssen Sie alle gegnerischen Steine schlagen.
Rückwärts (Ja/Nein)	Hier wird festgelegt, ob die Figuren rückwärts ziehen dürfen.
Zeitlimit	Wählen Sie hier Ja, können Sie ein Limit von 1-5 eingeben.
Sticks	Legen Sie hier die Anzahl der Joysticks fest (1 oder 2).

Wählen Sie mit Joystick in Port 0 und drücken Sie anschließend Fire.

Sebastian Mark

Listing

```
10 REM   Mark Sebastian's  KAH
Schuko Programing '86
Wingertstr.29/6700 Ludwigshafen
13 GOTO 3000
15 DIM TXT$(40):B1=9:B2=9:S=0:POKE 18,
0:POKE 19,0:POKE 20,0:POKE 16,64:POKE
53774,64
20 GRAPHICS 7:SETCOLOR 1,0,15:POKE 752
,1:GOSUB 5000
30 ? "K                KAH"
35 RESTORE 50:COLOR 2:PLOT 65,5
40 READ X,Y:IF X=-1 THEN 100
41 IF X=-2 THEN READ X,Y:PLOT X,Y:GOTO
 40
42 IF X=-3 THEN COLOR Y:GOTO 40
45 DRAWTO X,Y:GOTO 40
50 DATA 39,5,119,5,119,75,39,75,39,5,-
3,3,-2,0,40,12,118,12,-2,0,40,19,118,1
9,-2,0,40,26,118,26,-2,0,40,33,118,33
55 DATA -2,0,40,40,118,40,-2,0,40,47,1
18,47,-2,0,40,54,118,54,-2,0,40,61,118
,61,-2,0,40,68,118,68,-2,0
60 DATA 47,6,47,74,-2,0,55,6,55,74,-2,
0,63,6,63,74,-2,0,71,6,71,74,-2,0,79,6
,79,74,-2,0,87,6,87,74,-2,0
65 DATA 95,6,95,74,-2,0,103,6,103,74,-
2,0,111,6,111,74,-3,2,-2,0,39,19,62,19
,-2,0,96,19,118,19,-2,0
67 DATA 39,61,61,61,-2,0,97,61,118,61,
-2,0,55,27,55,32,-2,0,63,27,63,32,-2,0
,95,27,95,39,-2,0,103,27,103,32
70 DATA -2,0,71,34,71,39,-2,0,63,41,63
,53,-2,0,87,41,87,46,-2,0,47,48,47,53,
-2,0,95,48,95,53,-2,0,111,48,111,53
75 DATA -2,0,63,26,70,26,-2,0,80,26,86
,26,-2,0,71,33,79,33,-2,0,47,40,54,40,
-2,0,105,40,111,40,-2,0,80,47,87,47
80 DATA -2,0,72,54,78,54,-2,0,89,54,95
,54
99 DATA -1,0
100 REM FIGUREN AUFSTELLEN
101 R=9:IF KO=0 THEN R=10
105 COLOR 1:X=41:Y=10:FOR J=1 TO R:GOS
UB 1020:X=X+8:NEXT J:IF KO=1 THEN GOSU
B 1030
106 X=41:Y=73:IF KO=1 THEN GOSUB 1010:
X=X+8
110 FOR J=1 TO R:GOSUB 1000:X=X+8:NEXT
 J
115 FOR K=0 TO 500:NEXT K
120 REM HAUPTSCHLEIFE
121 IF TI<>0 THEN IF PEEK(18)>=TI THEN
 2100
125 X=41:IF S=0 THEN Y=73
126 IF S=1 THEN Y=10
130 ? "K        Spieler ";S+1:GOSUB 1050
135 P=S:IF JO=1 THEN P=0
140 IF STICK(P)=14 AND Y>10 THEN GOSUB
 1060:Y=Y-7:GOSUB 1050
142 IF STICK(P)=13 AND Y<73 THEN GOSUB
 1060:Y=Y+7:GOSUB 1050
144 IF STICK(P)=11 AND X>41 THEN GOSUB
 1060:X=X-8:GOSUB 1050
146 IF STICK(P)=7 AND X<113 THEN GOSUB
 1060:X=X+8:GOSUB 1050
148 IF STRIG(P)=1 THEN 140
150 GOSUB 1080:IF DEF=0 THEN GOSUB 107
0:GOTO 140
155 IF S=0 AND DEF>2 THEN GOSUB 1110:G
OTO 140
156 IF S=1 AND DEF<3 THEN GOSUB 1110:G
OTO 140
165 GOSUB 1060:SOUND 0,50,10,10:FOR J=
0 TO 30:NEXT J:SOUND 0,0,0,0
170 GO=0:IF STICK(P)=14 AND Y>10 THEN
GO=1
172 IF STICK(P)=13 AND Y<73 THEN GO=2
174 IF STICK(P)=11 AND X>41 THEN GO=3
176 IF STICK(P)=7 AND X<113 THEN GO=4
177 IF STRIG(P)=0 THEN 130
178 IF GO=0 THEN 170
179 IF RU=1 THEN IF GO=1 AND S=1 OR GO
=2 AND S=0 THEN 170
180 G=DEF*10+1000-10:ON GO GOSUB 200,2
50,300,350
185 IF S=0 THEN S=1:GOTO 120
190 IF S=1 THEN S=0:GOTO 120
200 REM FIGUR HOCH
201 LOCATE X+2,Y-5,M:IF M=2 THEN GOSUB
 1120:GOTO 170
205 Y=Y-7:GOSUB 1080:GOSUB 1130:GOSUB
G:Y=Y+7:GOSUB 1040:Y=Y-7:RETURN
250 REM FIGUR RUNTER
251 LOCATE X+2,Y+2,M:IF M=2 THEN GOSUB
 1120:GOTO 170
255 Y=Y+7:GOSUB 1080:GOSUB 1130:GOSUB
G:Y=Y-7:GOSUB 1040:Y=Y+7:RETURN
300 REM FIGUR LINKS
301 LOCATE X-2,Y-2,M:IF M=2 THEN GOSUB
 1120:GOTO 170
305 X=X-8:GOSUB 1080:GOSUB 1130:GOSUB
G:X=X+8:GOSUB 1040:X=X-8:RETURN
350 REM FIGUR RECHTS
```

```
351 LOCATE X+6,Y-2,M:IF M=2 THEN GOSUB
 1120:GOTO 170
355 X=X+8:GOSUB 1080:GOSUB 1130:GOSUB
G:X=X-8:GOSUB 1040:X=X+8:RETURN
1000 REM SPIELER 1- BAUER
1005 COLOR 1:PLOT X,Y:DRAWTO X+4,Y:PLO
T X+4,Y-1:PLOT X,Y-1:PLOT X+1,Y-2:PLOT
 X+3,Y-2:PLOT X+2,Y-3:RETURN
1010 REM SPIELER 1- KOENIG
1015 COLOR 1:PLOT X,Y:DRAWTO X+4,Y:PLO
T X,Y-1:DRAWTO X+4,Y-1:PLOT X+1,Y-2:DR
AWTO X+3,Y-2:PLOT X+2,Y-3:RETURN
1020 REM SPIELER 2- BAUER
1025 COLOR 1:PLOT X,Y-3:DRAWTO X+4,Y-3
:PLOT X,Y-2:PLOT X+4,Y-2:PLOT X+1,Y-1:
PLOT X+3,Y-1:PLOT X+2,Y:RETURN
1030 REM SPIELER 2- KOENIG
1035 COLOR 1:PLOT X,Y-3:DRAWTO X+4,Y-3
:PLOT X,Y-2:DRAWTO X+4,Y-2:PLOT X+1,Y-
1:DRAWTO X+3,Y-1:PLOT X+2,Y:RETURN
1040 REM FELD LOESCHEN
1045 COLOR 0:FOR J=Y TO Y-3 STEP -1:PL
OT X,J:DRAWTO X+4,J:NEXT J:RETURN
1050 REM SCANNER
1051 COLOR 2
1055 PLOT X-1,Y+1:DRAWTO X+5,Y+1:DRAWT
O X+5,Y-4:DRAWTO X-1,Y-4:DRAWTO X-1,Y+
1:RETURN
1060 REM CLEAR SCANNER
1065 COLOR 0:GOTO 1055
1070 REM ERROR SOUND
1075 SOUND 0,255,10,15:FOR J=0 TO 30:N
EXT J:SOUND 0,0,0,0:RETURN
1080 REM FELD UEBERPRUEFEN
1085 LOCATE X+2,Y,M:IF M=0 THEN DEF=0:
RETURN
1090 LOCATE X,Y,M:IF M=0 THEN 1100
1093 LOCATE X+2,Y-1,M:IF M=0 THEN DEF=
1:RETURN
1095 DEF=2:RETURN
1100 LOCATE X+2,Y-2,M:IF M=0 THEN DEF=
3:RETURN
1105 DEF=4:RETURN
1110 REM GEGNER!
1115 ? "K  Das ist eine feindliche Fig
ur!":GOSUB 1070:FOR J=0 TO 100:NEXT J:
? "K":RETURN
1120 REM BLOCKIERT
1125 ? "K       Der Weg ist blockiert!"
:GOSUB 1070:FOR J=0 TO 100:NEXT J:? "K
":RETURN
1130 REM WER MACHT WAS?
1132 IF DEF=0 THEN RETURN
1134 IF S=0 AND DEF<3 OR S=1 AND DEF>2
 THEN POP :GOSUB 1120:GOTO 170
1135 IF S=0 AND DEF=4 OR S=1 AND DEF=2
 THEN 2000
1140 IF S=0 AND DEF=3 THEN GOSUB 1150:
RETURN
1145 IF S=1 AND DEF=1 THEN GOSUB 1160:
RETURN
1146 STOP
1150 REM BAUER 2 -GEKILLT
1155 B2=B2-1:IF B2=0 THEN 2000
1156 SOUND 0,100,8,10:GOSUB 1040:FOR J
=0 TO 5:NEXT J:SOUND 0,0,0,0:RETURN
1160 REM BAUER 1 -GEKILLT
1165 B1=B1-1:IF B1=0 THEN 2000
1166 GOTO 1156
2000 REM SIEG
2005 IF S=0 THEN X=10:Y=20
2010 IF S=1 THEN X=135:Y=20
2015 COLOR 2:PLOT X+2,Y:DRAWTO X+10,Y:
DRAWTO X+10,Y-10:DRAWTO X+8,Y-7:DRAWTO
 X+6,Y-10:DRAWTO X+4,Y-7:DRAWTO X+2,Y-
10:DRAWTO X+2,Y
2020 ? "K Spieler ";S+1;" hat gewonnen
!-Fire.."
2025 IF STRIG(0)=1 AND STRIG(1)=1 THEN
 2025
2030 CLR :RUN
2100 IF B1>B2 THEN S=0:? "K":GOTO 2005
2105 IF B2>B1 THEN S=1:? "K":GOTO 2005
2110 IF B1=B2 THEN ? "K  Zeit abgelauf
en - unentschieden!":FOR X=1 TO 150:NE
XT X
2115 ? "K  &+ Fire..":IF STRIG(0)=1 AN
D STRIG(1)=1 THEN 2115
2120 CLR :RUN
3000 GRAPHICS 0:SETCOLOR 4,0,15:SETCOL
OR 2,0,15:SETCOLOR 1,0,0:CLR :POKE 752
,1
3001 RU=0:KO=1:TI=0:JO=2:PIC=9
3005 ? :? :? ,"          KAH":? :? :? :? :
? :?
3010 ? " Rueckwaerts JA:NEIN"
3011 ? " Koenige     JA:NEIN"
3012 ? " Zeitlimit   JA:NEIN  - 0"
3013 ? " Sticks      2"
3014 POSITION 10,21:? "Von Mark Sebast
ian"
3015 POSITION 2,PIC:? "&+":L=0:R=0
3016 IF STRIG(0)=0 THEN 15
3020 IF STICK(0)=14 AND PIC>9 THEN POS
ITION 2,PIC:? " ":PIC=PIC-1:GOTO 3015
3022 IF STICK(0)=13 AND PIC<12 THEN PO
SITION 2,PIC:? " ":PIC=PIC+1:GOTO 3015
3024 IF STICK(0)=15 THEN FOR J=0 TO 5:
NEXT J:GOTO 3015
3026 IF STICK(0)=11 THEN L=1
3028 IF STICK(0)=7 THEN R=1
3030 IF L=1 AND PIC=9 THEN POSITION 15
,9:? "JA:NEIN":RU=0:GOTO 3015
3032 IF R=1 AND PIC=9 THEN POSITION 15
,9:? "JA:NEIN":RU=1:GOTO 3015
3034 IF L=1 AND PIC=10 THEN POSITION 1
5,10:? "JA:NEIN":KO=1:GOTO 3015
3036 IF R=1 AND PIC=10 THEN POSITION 1
5,10:? "JA:NEIN":KO=0:GOTO 3015
3038 IF L=1 AND PIC=11 THEN POSITION 1
5,11:? "JA:NEIN":GOTO 3050
3040 IF R=1 AND PIC=11 THEN POSITION 1
5,11:? "JA:NEIN  - 0":TI=0:GOTO 3015
3042 IF L=1 AND PIC=12 THEN POSITION 1
5,12:? "1":JO=1:GOTO 3015
3044 IF R=1 AND PIC=12 THEN POSITION 1
5,12:? "2":JO=2:GOTO 3015
3046 GOTO 3015
3048 GOTO 15
3050 POSITION 26,11:? TI
3055 IF STICK(0)=14 OR STICK(0)=13 THE
N 3015
3060 IF STICK(0)=11 AND TI>1 THEN TI=T
I-1:GOTO 3050
3065 IF STICK(0)=7 AND TI<5 THEN TI=TI
+1:GOTO 3050
3070 GOTO 3050
5000 REM DL-MANIPULATION
5010 DL=PEEK(560)+256*PEEK(561)
5020 POKE DL+6,2:POKE DL+32,13:RESTORE
 5060:POKE DL+88,13:POKE DL+89,13:POKE
 DL+90,13
5030 READ TXT$:T=LEN(TXT$):POKE 87,0
5040 FOR I=1 TO T:COLOR ASC(TXT$(I,I))
5050 PLOT I,1:NEXT I
5060 DATA             Mark Sebastian's  KA
H
5070 POKE 87,7:RETURN
5100 REM KAH ist ein aelteres Spiel vo
n mir,deswegen auch diese etwas seltsa
me Struktur.Ihr koennt das Spiel
5105 REM selbstverstaendlich verbesser
n.Wenn ihr gute Verbesserungen habt,ko
ennt ihr sie mir ruhig schicken!
```

Memorix – ein neues Spiel für den ST

Diesem Programm liegt eigentlich eine sehr alte Idee zugrunde, die sich am Computer mit GFA-Basic allerdings herrlich ausbauen läßt: Memory, das bekannte Spiel unserer Kindheit.

Spielanleitung

Eine Serie quadratischer Karten wird mit der Bildseite nach unten auf dem Tisch ausgebreitet. Der Reihe nach deckt nun jeder Mitspieler zwei Karten auf. Stimmen die Motive überein, so darf er das Paar behalten und weiter aufdecken. Um zu gewinnen, ist neben etwas Glück ein gutes Gedächtnis notwendig.

16 Bit

Bei "Memorix", der Computerversion dieses Spiels, können entweder zwei Personen gegeneinander oder ein Einzelner gegen den Computer antreten. 14 Karten (7 Paare) werden ausgeteilt. Spieler 1 beginnt, indem er zwei beliebige Karten durch Anklicken umdreht. Beim Programmstart wird ein Kartensatz aus den Buchstaben A bis G erzeugt, jeweils mit einem zufälligen Rahmen umgeben.

Fungiert der Computer als Gegenspieler, so muß man sein Gedächtnis schon sehr anstrengen, da ein spezieller Spielalgorithmus programmiert wurde. Von jeder Karte, die bereits umgedreht war (und wirklich nur von solchen) merkt sich der Computer die Position. Ist er dann am Zug, so überprüft er zuerst, ob ihm bereits ein Paar bekannt ist, und deckt dieses dann auf. Ist kein komplettes Paar gespeichert, wird eine unbekannte Karte gewählt. Ist hierzu das Doppel bereits erschienen, wird umgedreht. Wenn kein Paar zu ermitteln ist, so wird auch die zweite Karte zufällig ausgesucht. Sollte der Computer auf diese Weise zu gut spielen, kann man in der Routine COMPUTER_MOVE eventuell auch ein "vergeßliches Gedächtnis" programmieren, das zufällige Werte einfach löscht.

Menü

Nach genügend Niederlagen gegen den Computer sollte man "Memorix" nicht gleich zur Seite legen, denn die interessantesten Möglichkeiten bietet das Menü Karten. EDITIEREN öffnet zwei Fenster, mit deren Hilfe eigene Kartensätze erstellt werden können. Im kleineren, linken Fenster kann mit den Up/Down-Arrows das zu bearbeitende Teil eingestellt werden. Indem man das Fenster dann auf die große "Memorix"-Lupe zieht, wird es um den Faktor 3 vergrößert und kann nun von Hand geändert werden. Mit der linken Maustaste werden dabei Punkte gesetzt, mit

Memorix Spiel Karten Player: 2 Punkte: 3 : 0

der rechten gelöscht. Hat man FILL aktiviert, so wird mit den entsprechenden Farben gefüllt.

Wer seinen künstlerischen Fähigkeiten mißtraut, kann mit der Funktion BILD LADEN jedes "Degas"-Monochrom-Bild laden und daraus ein Teil ausschneiden. Wie man "Memorix" an andere Bildformate anpaßt, war im ersten **ATARI** *magazin* beim "Puzzler" zu lesen.

Ist man mit seiner Arbeit zufrieden, wird die Kreation in den Teilesatz übernommen, indem man das große Fenster über das kleine zieht. Jetzt kann sie abgespeichert und später wieder geladen und verändert werden. Als Extender wird dabei automatisch .MIX verwendet.

Mit dem Closer kann der Inhalt des jeweiligen Fensters auch gelöscht werden, was aber normalerweise nur in der Lupe sinnvoll ist. Für besondere Effekte kann man auch im Karte-Window malen; allerdings muß das Kunstwerk erst vergrößert und dann wieder zurückkopiert werden, um es wirklich zu ändern. Für besonders schwere Kartensätze können Teile, die sich nur geringfügig unterscheiden, so auch kopiert werden. Will man den Inhalt der Lupe invertieren, so ist dies mit der Full-Window-Box möglich.

Prozeduren

Indem wir nun einige Prozeduren etwas näher betrachten, will ich versuchen, das Programm leichter verständlich zu machen.

MOUSE_SELECT

Hier wird gewartet, bis eine Maustaste gedrückt ist, um dann die Koordinaten des Zeigers zu ermitteln und daraus das gewünschte Teil zu berechnen. Dieses wird nur dann umgedreht, wenn im Feld PP( ) noch kein TRUE eingetragen ist. Gleichzeitig wird die Feldnummer aus PP in den Computer Memory Cm( ) übernommen, um für die Zugberechnung zur Verfügung zu stehen.

MESAUSWERT (Message-Auswertung)

Hier werden alle Manipulationen an den Fenstern festgestellt und entsprechend darauf reagiert. Natürlich können hier nur die Funktionen ankommen, die bei mindestens einem der Windows definiert sind. Ebenfalls muß ein ON MENU MESSAGE GOSUB MESAUSWERT ausgeführt worden sein. Beim nächsten ON MENU wird – sofern eine Message vorliegt – diese Routine angesprungen. Hier nun die verwendeten Funktionen:



In Menü (4) steht immer die Nummer des Windows, in/mit dem die Funktion aktiviert wird.

Window moved: Menü (1) wird 28, in Menü (5) und (6) stehen die Koordinaten, an die das Window bewegt wurde.

Full Window: Menü (1) ist 23.

Close Window: Menü (1) ist 22.

Window topped: Menü (1) ist 21.

Arrow up/down: Menü (1) ist 24. Menü (5) enthält die Kennung des verwendeten Pfeils.

MAKE WINDOW

Hier werden die beim Funktionsaufruf übergebenen Werte in die Window-Tabelle (WINDTAB) des GFA-Basic eingetragen. Da jeder Eintrag aus 12 Byte besteht, errechnet sich der Faktor für die Fensternummer, in der die Werte gelten sollen, aus dem Produkt der Window-Nummer mit 12.

Beim vorliegenden Programm wurden übrigens erste Versuche mit der GFA-Basic-Version 2.0 angestellt. Soll es für Version 1.x verwendet werden, so ist die LOAD/SAVE-Routine zu ändern und am Anfang des Programms die Konstante TRUE mit −1 zu definieren (wichtig!).

Wie immer sind wir auch hier auf Reaktionen unserer Leser sehr gespannt. Sollten Ihnen besonders schöne Kreationen gelingen, so werden wir diese in einer der folgenden Ausgaben des **ATARI** ***magazins*** gerne veröffentlichen.

Thomas Tausend

Listing in GFA-Basic

```
  ************************************************************
**********************************
  * Memorix - ein Megabrain-Produkt des Allgäu-Tea
ms * (c) by Thomas Tausend *
  ************************************************************

Menu Kill
Closew 1
Closew 2
  Flags für die Window-Verwaltung
Let Name=1
Let Close=2
Let Full=4
Let Move=8
Let Info=16
Size=32
Uparrow=64
Dnarrow=128

Dim M$(58),P$(14),Pp(28),Cm(28),Punkte(2)
Arrayfill Pp(),True
Arrayfill Cm(),True
Mentxt:
Data Memorix,  Memorix-Info,--------------------,
1,2,3,4,5,6,""
Data Spiel,  Mensch-Compi,  Mensch-Mensch,--------
--------,  starten
Data -----------------,  beenden,""
Data Karten,  einladen,  abspeichern,--------------
--,  editieren
Data ----------------,  Bild laden,  ausschneiden,
 fill,"","",tendet
Do
  Read M$(Z)
  Exit If M$(Z)="tendet"
  Inc Z
Loop
@Create_default
Menu M$()
On Menu  Gosub Menauswert
Menu 11,1
Menu 24,2
Menu 25,2
Menu 26,2
Cflag=True
Eflag=True
Do
  On Menu
Loop
End
Procedure Menauswert
  Menu Off
  H$=M$(Menu(0))
  If H$="  Memorix-Info"
    @Show_about
  Endif
  If H$="  Mensch-Compi"
   Cflag=True
   Menu 11,1
   Menu 12,0 ! Haken löschen
 Endif
 If H$="  Mensch-Mensch"
   Cflag=0
   Menu 11,0
   Menu 12,1
 Endif
 If H$="  beenden"
   End
 Endif
 If H$="  starten"
   Eflag=True
   Menu 22,3
   @Play_game
 Endif
 If H$="  Bild laden"
   Sget A$
   Fileselect "\*.pi3","",F$
   If F$<>""
```

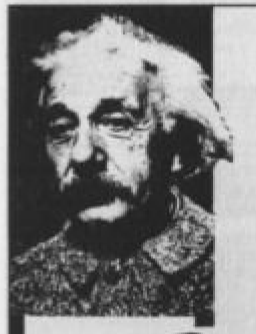
```
      Bload F$,Xbios(2)-34! Degas-Format
      Sget B$
      Menu M$()! Menueleiste wieder ein
      Menu 11,1
      Menu 26,-Fflag
      Menu 22,-Eflag+2
    Endif
    Sput A$
  Endif
  If H$="  einladen"
    Fileselect "\*.mix","",F$
    If F$<>""
      Open "I",#1,F$
      For N=0 To 13
        Bget #1,Varptr(P$(N)),806
      Next N
      Close
      If Eflag=False
        Put 51,87,P$(Tn)
        Infow 2,F$
      Endif
    Endif
  Endif
  If H$="  ausschneiden"
    Closew 0
    Sget A$
    Sput B$
    @Create_parts
    Sput A$
    Put 51,87,P$(Tn)
    Openw 1
  Endif
  If H$="  editieren"
    Eflag=Not Eflag
    Menu 22,-Eflag+2
    Menu 24,3
    Menu 26,3
    @Edit
  Endif
  If H$="  fill"
    Fflag=Not Fflag
    Menu 26,-Fflag
  Endif
  If H$="  abspeichern"
    Fileselect "\*.mix","",F$
      If F$<>""
        If Right$(F$,3)<>"mix"
          F$=F$+".mix"
        Endif
        Open "O",#1,F$
        For N=0 To 13
          Bput #1,Varptr(P$(N)),806
        Next N
        Close #1
        Tz=0
      Endif
    Endif
  Return
  Procedure Show_about
    Alert 1,"M e m o r i x|ein Megabrain-Produkt|von
   Thomas Tausend|für das ATARI-Magazin",1," Aha ",D
  ummy
  Return
  Procedure Play_game
    Closew 0
    Arrayfill Punkte(),0
    @Paint_backs
    @Mix_cards
    Repeat
      For Player=1 To 2
        Print At(40,1);"Player: ";Player
        Repeat
          If Player=2 And Cflag=True
            @Computer_move
          Else
            @Mouse_select
            Z1=Z
            S1=S
            While S1=S And Z1=Z
              @Mouse_select
            Wend
          Endif
          If Pp(S1+Z1*7)=Pp(S+Z*7)
            Pp(S+Z*7)=True
            Pp(S1+Z1*7)=True
            Cm(S+Z*7)=True
            Cm(S1+Z1*7)=True
            Alert 1,"Übereinstimmung!| |noch ein Ver
such!",1,"Gut",Dummy
            Inc Punkte(Player)
            Print At(55,1);"Punkte: ";Punkte(1);" :
";Punkte(2);" "
            Deffill 0
            Pbox S*90+10,Z*90+30,S*90+90,Z*90+110
            Pbox S1*90+10,Z1*90+30,S1*90+90,Z1*90+11

          Else
            While Mousek=0 And Inkey$=""
            Wend
            While Mousek<>0 Or Inkey$<>""
            Wend
            @Turn(S1,Z1)
            @Turn(S,Z)
          Endif
          Exit If Punkte(1)+Punkte(2)=14
        Until Pp(S1+Z1*7)<>Pp(S+Z*7)
        Exit If Punkte(1)+Punkte(2)=14
      Next Player
    Until Punkte(1)+Punkte(2)=14
    Print At(40,1);"Endstand: "
    Alert 0,"ENDE|Ich hoffe|M e m o r i x|hat Ihnen
    gefallen!",1," Danke ",Dummy
    Return
    Procedure Create_default
      Deftext 1,16,0,32
      For N=0 To 13
        Deffill 1,2,Random(23)+1
        Pbox 0,100,79,179
        Deffill 0
        Pbox 20,120,60,160
        Text 33,150,Chr$(N+65)
        Get 0,100,79,179,P$(N)
      Next N
      Pbox 0,100,80,180
    Return
    Procedure Paint_backs
      Deffill 0
      Pbox 0,19,640,400 ! Bildschirm löschen
      Deffill 1,2,16
      For Z=0 To 3
        For S=0 To 6
          Pbox S*90+10,Z*90+30,S*90+90,Z*90+110
        Next S
      Next Z
    Return
    Procedure Mix_cards
      For N=0 To 13 ! 14 versch. Teile
        For M=0 To 1 ! je 2 Bilder
          Repeat
            H=Random(28)
          Until Pp(H)=True
          Pp(H)=N
        Next M
      Next N
      Return
      Procedure Mouse_select
        Repeat
          Repeat
            While Mousek=0
            Wend
            S=Int((Mousex)/90)
            Z=Int(Abs((Mousey-25))/90)
          Until S<8 And Z<4
          If Pp(S+Z*7)<>True
            Cm(S+Z*7)=Pp(S+Z*7)
            Put S*90+10,Z*90+30,P$(Pp(S+Z*7))
          Endif
          While Mousek=1
          Wend
        Until Pp(S+Z*7)<>True
      Return
      Procedure Turn(S,Z)
        Deffill 1,2,16
        Pbox S*90+10,Z*90+30,S*90+90,Z*90+110
      Return
      Procedure Computer_move
        M1=True ! Noch kein Zug gefunden
        M2=True !  "    "    "     "
        For P=0 To 25 ! Zuerst nach bekanntem
          For Ps=P+1 To 27 ! Paar suchen...
            If Cm(Ps)=Cm(P) And Cm(P)<>True
              M1=P ! Wenn gefunden
              M2=Ps ! beide merken
            Endif
          Next Ps
        Next P
        If M1=True ! Wenn kein Paar gefunden
       Repeat ! eine neue Karte
         M1=Random(28) ! zufällig umdrehen
       Until Cm(M1)=True And Pp(M1)<>True
       Cm(M1)=Pp(M1) ! Nach zugehörigem
       For P=0 To 27 ! Doppel im Gedächtnis
         If Cm(P)=Cm(M1) And M1<>P ! suchen
           M2=P ! Gefunden? - Merken!
         Endif
       Next P
     Endif
     If M2=True ! Wenn kein Doppel bekannt
       Repeat !   zufällige Karte umdrehen
         M2=Random(28)
       Until Cm(M2)=True And Pp(M2)<>True
       Cm(M2)=Pp(M2)
     Endif
     Z=Int(M1/7)
     S=M1-Z*7
     Z1=Int(M2/7)
     S1=M2-Z1*7
     Put S*90+10,Z*90+30,P$(Pp(S+Z*7))
     Put S1*90+10,Z1*90+30,P$(Pp(S1+Z1*7))
   Return
   Procedure Create_parts
     While Mousek<>False
     Wend
     Graphmode 3
     Repeat
       Mouse X,Y,K
       Pbox X,Y,X+79,Y+79
       While Mousex=X And Mousey=Y And Mousek=False
       Wend
       Pbox X,Y,X+79,Y+79
     Until K<>False
     Get X,Y,X+79,Y+79,P$(Tn)
     Put 51,87,P$(Tn)
   Return
   Procedure Edit
     On Menu Message Gosub Mesauswert
```

```
  Deffill 0
  Pbox 0,19,640,400 ! Bildschirm löschen
  Graphmode 0
  Titlew 1," Karte "
  Infow 1," Nummer: 0"
  @Make_window(1,Name+Close+Move+Info+Uparrow+Dnar
row,50,50,100,110)
  Put 51,87,P$(0)
  Titlew 2," Memorix-Lupe (c) TT86 "
  Infow 2," (c) by Thomas Tausend"
  @Make_window(2,Name+Close+Full+Move+Info,238,30,
243,279)

  Do
    On Menu
    Mouse X,Y,K
    If K<>0
      Deffill -1*(K-2),2,8
      Xpos=Int(X/3)*3
      Ypos=Int(Y/3)*3
      If Fflag=True
        Fill X,Y
      Endif
      If X>0 And X<241 And Y>0 And Y<241
        Pbox Xpos,Ypos,Xpos+3-1,+Ypos+3-1
      Endif
    Endif
  Loop
  Return
  Procedure Do_small
    Closew 0
  For Z%=0 To 79
    For S%=0 To 79
      Color Point(S%*3+240,Z%*3+60)
      Draw 51+S%,87+Z%
    Next S%
  Next Z%
  Get 51,87,131,167,P$(Tn)
  Openw 1
Return

Procedure Do_big
  Closew 0
  For Z%=0 To 79
    For S%=0 To 79
      Deffill Point(51+S%,87+Z%),2,8
      Pbox S%*3+239,Z%*3+67,S%*3+241,Z%*3+69
    Next S%
  Next Z%
  Openw 2
Return
Procedure Mesauswert
  Winum=Menu(4)
  If Menu(1)=20
    If Winum=2 And Menu(5)<100 And Menu(6)<100
      @Do_small
    Endif
    If Winum=1 And Menu(5)>230 And Menu(6)>30
      @Do_big
    Endif
  Endif
  If Menu(1)=23
    Graphmode 3
    Pbox -1,-1,240,240
    Graphmode 0
  Endif
  If Menu(1)=22
    Clearw Winum
  Endif
  If Menu(1)=21
    Openw (Winum)
  Endif
  If Menu(1)=24
    If Menu(5)=3 And Tn<13
      Inc Tn
    Endif
    If Menu(5)=2 And Tn>0
      Dec Tn
    Endif
    Infow 1," Nummer: "+Str$(Tn)
    Put 51,87,P$(Tn)
  Endif
Return
Procedure Make_window(Winum,Flags,Wix,Wiy,Wib,Wih)
  Local Faktor
  Faktor=(Winum-1)*12
  Dpoke Windtab+2+Faktor,Flags
  Dpoke Windtab+4+Faktor,Wix
  Dpoke Windtab+6+Faktor,Wiy
  Dpoke Windtab+8+Faktor,Wib
  Dpoke Windtab+10+Faktor,Wih
  Openw Winum
Return
```



1000,- TOP PROGRAMM DES MONATS

"Star-Castle" heißt das Topprogramm in diesem Heft. Es wurde auf einem Atari 800 geschrieben, läuft aber auch auf allen 8-Bit-Ataris mit 48 KByte. Die Aufgabe besteht darin, angreifende Roboter abzuwehren, die den Zentralcomputer durch Erschütterung zerstören wollen. Nach jedem Level wird das Spiel schneller.

Der Autor dieses Spiels ist Dirk Jacobasch. Er ist 18 Jahre alt und besucht die Fachoberschule im Bildungsgang Elektronikassistent zur Erlangung der Fachhochschulreife. Später will er dann Elektronik oder Informatik studieren. Dirk programmiert neben Basic in Assembler und beschäftigt sich am liebsten mit Grafik. Als weiteres Hobby treibt er Sport, speziell Surfen und Schwimmen.



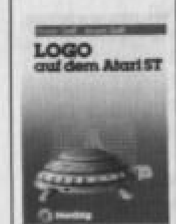
ATARI



Lüke/Lüke
Der ATARI 520 ST

Dies ist das Buch für die erste Begegnung mit dem Atari ST. Die zweite Auflage dieses Standardwerks wurde überarbeitet und berücksichtigt nun die Neuerungen beim Betriebssystem und den Systemprogrammen. Daher wird diese allgemeine Beschreibung der Hard- und Software auch für die tägliche Arbeit ein hilfreicher Ratgeber sein.

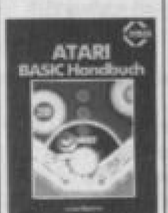
Bestellnummer MT 23 **DM 49.–**



Geiß/Geiß
Logo auf dem Atari ST

Die Programmiersprache Logo erfreut sich steigender Beliebtheit. Nicht zuletzt deshalb, weil sie zum Lieferumfang des ST gehört. Daß Logo nicht nur für grafische Spielereien taugt, wird mit diesem Buch bewiesen: Textverarbeitung oder Mathematik sind ebenso möglich.

Bestellnummer HU 1 **DM 35.–**



Julian Reschke
Atari Basic Handbuch

208 Seiten
Das vorliegende Basic-Handbuch hilft Ihnen, Ihren Atari voll und ganz zu beherrschen. Das vollständige Basic-Vokabular wird beschrieben und anhand praktischer Beispiele erläutert.

Bestellnummer SY 13 **DM 32.–**



D. Senftleben
Start mit Atari-Logo

220 Seiten
Hier handelt es sich um eine benutzerfreundliche Einführung in die Computersprache Logo. Grafik, Text und Musik werden in zwölf Lektionen besprochen. Auch große Bildschirmfotos fehlen nicht. Die Atari-Logo-Vokabeln, die im Buch aufgeführt sind, erschließen dem Leser neue Einsatzbereiche.

Bestellnummer V 2 **DM 30.–**



L. M. Schreiber
Das Atari-Programmierhandbuch

390 Seiten
Hier werden keinerlei Kenntnisse vorausgesetzt. Sie lernen den Weg vom Problem zum Programm (einschließlich Flußdiagramm und dessen Gebrauch). Außerdem wird erklärt, wie Sie den 6502-Prozessor direkt programmieren. Wenn Sie dieses Buch durchgearbeitet haben, kennen Sie Ihren Atari in- und auswendig.

Bestellnummer MT 8 **DM 52.–**



Raabe/Schmidt
Spielen, lernen und arbeiten mit dem Atari

280 Seiten
Damit werden Ihnen theoretische und praktische Kenntnisse vermittelt. Von Anfang an lernen Sie Ihren Rechner Schritt für Schritt immer besser kennen und beherrschen. So werden Sie vom Spieler zum Profi.

Bestellnummer SY 14 **DM 32.–**



A. Hettinger/A. Heinz
Start mit Atari-Basic

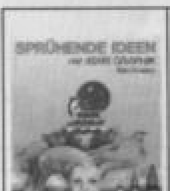
184 Seiten
Nach dem Durcharbeiten dieses Buches werden Sie selbst in der Lage sein, Programme zu schreiben. Angefangen bei Grafik- und Soundmöglichkeiten über Tips und Tricks bis hin zu kompletten Spielprogrammen reicht das breite Spektrum. Neben dem eigentlichen Basic-Kurs bildet die komplett dokumentierte Liste aller Atari-Basic-Befehle die Krönung des Ganzen.

Bestellnummer V 3 **DM 30.–**

Dittrich
ATARI-ST – Peeks & Pokes

Wem Programmiersprachen und Anwenderprogramme auf dem ST nicht genügen, der ist mit diesem Buch gut bedient. Hier wird man hinter die Kulissen dieses außerordentlichen Computers geführt. Denn nur so können die wahren Dimensionen des Atari ST ausgelotet werden.

Bestellnummer DB 30 **DM 29.–**



Tom Rowley
Sprühende Ideen mit Atari Grafik

250 Seiten
Dies ist ein Lehrbuch, das mit den Grafikmöglichkeiten des Atari in die Gestaltgesetze von Objekten, in Farbgebung und in die Entwicklung von Bildschirmentwürfen einführt.

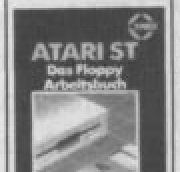
Bestellnummer TW 15 **DM 49.–**



A. + J. Peschetz
Was der Atari alles kann
Band 1

236 Seiten
Hier muß der Anwender schon die Grundbegriffe des Atari-Basic kennen und ein wenig Übung im Programmieren besitzen. Eine Vielzahl von gut durchstrukturierten Programmen aus den Bereichen Hobby, Wissenschaft, Beruf und Spiel werden vorgestellt.

Bestellnummer V 4 **DM 35.–**



Aumann/Maier/Stöpper
ATARI ST – Das Floppy-Arbeitsbuch

Die Diskettenstationen SF 354 und SF 314 sind Thema dieses Buches. Ausführlich wird auf die Programmierung im Diskettenbetrieb eingegangen. Beigelegt ist zusätzlich eine Diskette mit leistungsfähigen Programmen. So z.B. ein Kopierprogramm oder ein Disketten-Monitor.

Bestellnummer SY 29 **DM 69.–**



Naimann
ATARI-ST – Einführung in WordStar

WordStar gilt nach wie vor als der Klassiker der Textverarbeitung. Daher ist dieses Programm unter CP/M 2.2 auch für den Atari ST verfügbar. Dieses Buch ist eine Einführung in die Arbeit mit WordStar und wird dem Fortgeschrittenen als unentbehrliches Nachschlagewerk dienen.

Bestellnummer SY 30 **DM 48.–**



A. + J. Peschetz
Was der Atari alles kann
Band 2

240 Seiten
Entsprechend Band 1 enthält auch dieses Buch eine ausgewogene Mischung aus professionellen Anwendungsprogrammen und Spielen wie z.B. Dateiorganisation, Datensortiermethoden aber auch Trigonometrie in Verbindung mit deren ausgeklügelten Erläuterungen.

Bestellnummer V 5 **DM 35.–**



Stanley R. Trost
Atari-Programm-Sammlung

190 Seiten
Hier wird dem Anwender ein Satz ausgetesteter Programme für die Atari-Computer geboten. Eine breite Palette praktischer Beispiele hilft Ihnen, Ihren Computer optimal zu nutzen.

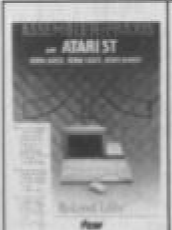
Bestellnummer SY 11 **DM 34.–**



Voss
Das Basic-Trainingsbuch zu Atari 600 XL/800 XL

383 Seiten
Das Basic-Trainingsbuch zu Atari 600 XL/800 XL ist eine ausführliche, didaktisch gut geschriebene Einführung in das Atari-Basic. Von den Befehlen über die Problemanalyse bis zum fertigen Algorithmus lernt man schnell das Programmieren.

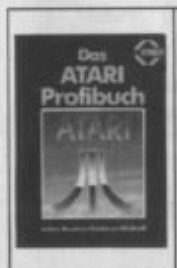
Bestellnummer DB 17 **DM 39.–**



Löhr
Assembler-Praxis auf Atari ST

Die Programmierung des 68000-Mikroprozessors auf dem Atari ST verlangt nicht nur genaue Kenntnisse des System, auch der Umgang mit Assemblern oder Editoren will gelernt sein. Diesen Gesichtspunkt der praktischen Anwendung stellt der Autor des Buches in den Mittelpunkt, wobei auch die Grundlagen keineswegs zu kurz kommen.

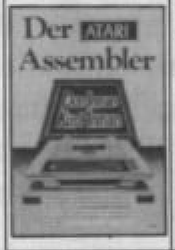
Bestellnummer TW 16 **DM 59.–**



Reschke/Wiethoff
Das Atari Profibuch

320 Seiten
In diesem Werk finden Sie gebündelt alle wichtigen Informationen, um Ihren Atari genau kennenzulernen und seine Fähigkeiten voll auszureizen. Ein Informationspaket, das keine Fragen offen läßt.

Bestellnummer SY 12 **DM 42.–**



Don Inman/Kurt Inman
Der Atari Assembler

276 Seiten
Mit diesem Buch können Sie das Programmieren in Assembler lernen und sich gleichzeitig mit der Anwendung des Atari-Assembler-Moduls auf Ihrem Atari 400- oder 800-Modell vertraut machen.

Bestellnummer ID 18 **DM 36.–**

BUCHVERSAND



Grohmann/Seidler/Silbar
Das Maschinensprachebuch zum ATARI ST

Der Mikroprozessor 68000 verleiht dem Atari ST seine sprichwörtliche Leistungsfähigkeit. Dieses Buch ist ein Lehrbuch zur Programmierung in Maschinensprache. Auch ohne Vorkenntnisse bekommt man damit bald direkten Zugang zu den unglaublichen Fähigkeiten dieses Prozessors.

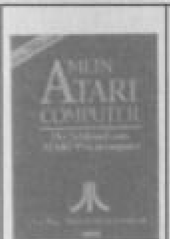
Bestellnummer DB 28 DM 39.–



C. Lorenz
Das große Spielebuch für Atari, Band 1

151 Seiten
Aufregende Computerspiele in Atari-Basic. Neben Spielen finden Sie hier eine Reihe hochinteressanter Anregungen für eigene Programme. 3-D-Grafik, Bewegung und Scrollen, Grafik und Ton in Forth, Tonprogrammierung usw.

Bestellnummer HO 25 DM 29.80



Poole/McNiff/Cook
Mein Atari-Computer

500 Seiten
Ein Handbuch, das für jeden Atari-Besitzer wertvolle Informationen enthält und zur Lösung aller Atari-Probleme beiträgt. Es ist reich bebildert und enthält eine Vielzahl der für den ernsthaften Interessierten so wichtigen Tabellen.

Bestellnummer TW 20 DM 59.–



C. Lorenz
Das große Spielebuch für Atari, Band 2

200 Seiten
Dieses Buch enthält Programme für den Atari 600 XL/800 XL und ist eine Weiterführung von Band 1. Es bringt eine Reihe neuer Spiele, Programme zur Sounderzeugung und ein Kapitel über Grafik-Spielereien.

Bestellnummer HO 26 DM 29.80



Steiner/Steiner
GEM für den Atari 520 ST

344 Seiten
Dieses Werk ist eine Einweisung in alle Bereiche, die GEM für den Benutzer interessant machen. Der unerfahrene Anwender findet eine Menge Tips für die Bedienung, um effektiv mit dem Atari ST arbeiten zu können.

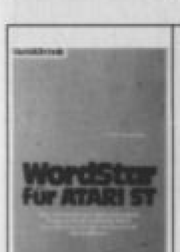
Bestellnummer MT 21 DM 52.–



Walkowiak
Adventures, und wie man sie auf dem Atari 600 XL/800 XL programmiert

284 Seiten
Hier wir gezeigt, wie Adventures funktionieren, wie man sie erfolgreich spielt, und wie man eigene Adventures auf Atari-Computern der Serie XL programmiert. Hinzu kommt ein kompletter Adventure-Generator, der das Selberprogrammieren zum Kinderspiel macht.

Bestellnummer DB 27 DM 39.–



Jürgensmeier
WordStar für den Atari ST

435 Seiten
Dieses Buch ist so aufgebaut, daß der Leser mit WordStar schrittweise vertraut wird. Anhand von Beispielen werden alle Funktionen erläutert. Auch auf die Bedienung von MailMerge wird ausführlich eingegangen.

Bestellnummer MT 22 DM 49.–



Schwaiger
Atari Star-Texter

110 Seiten + Disk
Hierbei handelt es sich um eine umfangreiche, komfortable Textverarbeitung für Ihren Atari (mind. 48 KByte). Das Buch gibt eine Einführung, die Diskette bietet ein exzellentes Programm.

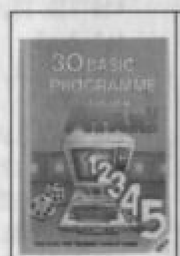
Bestellnummer SY 28 DM 64.–



Severin
Das große DFÜ-Buch zum ATARI ST

Die Verbindung des Atari ST mit der weiten Welt der Mailboxen und Datenbanken wird in diesem Buch mit allen Aspekten abgehandelt. Von den Grundlagen über ein komplettes Mailboxprogramm bis zur Telefonliste bietet es eine fundierte Einführung in die DFÜ.

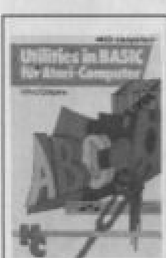
Bestellnummer DB 29 DM 39.–



Rugg/Feldman/Barry
30 Basic-Programme für den Atari

274 Seiten
Das Buch enthält sorgfältig getestete Spiel- und Grafikprogramme aus Mathematik, Unterricht und vielen anderen Anwendungsbereichen des täglichen Lebens für Ihren Atari-Computer.

Bestellnummer ID 29 DM 34.–



Alfred Görgens
Utilities in Basic für Atari-Computer

120 Seiten
In diesem Buch finden Sie praktische Utilities zu den Themen Programmierhilfe, Sound und Textverarbeitung. So z.B. automatische Zeilennumerierung, Umnumerierung von Basic-Zeilen, automatischer Programmstart, Musikeditor oder auch die Wiedergabe von Atari-Zeichen und Musiknoten auf dem Drucker.

Bestellnummer V 24 DM 25.–



James/Gee/Ewbank
Das Atari Spielebuch für 600 XL/800 XL

184 Seiten
21 Spiele vermitteln Spannung, Action und bewegte Grafik. Jedes Programm ist vollständig aufgelistet und ausführlich erläutert. So wird auch der Newcomer mit der Syntax und dem Aufbau der Programmiersprache Basic vertraut gemacht. Er lernt die Routine verstehen, analysieren und kann sie somit auch in eigene Programme einbinden.

Bestellnummer V 30 DM 30.–



Buch-Bestellschein

Bitte liefern Sie mir folgende Bücher:

Anzahl	Bestell-Nr.	Einzel-Preis

Ich wünsche folgende Bezahlung:

☐ Nachnahme (+ 5.70 DM Porto + Versandkosten)

☐ Vorauskasse (keine Versandkosten)

Bei Vorauskasse bitte Scheck beilegen oder auf Postscheckkonto Karlsruhe 43423-756 überweisen.

Name des Bestellers

Anschrift

PLZ/Ort

Telefon

Datum/Unterschrift

Coupon ausschneiden, auf Postkarte kleben und einsenden: ATARI magazin, Postfach 1640, 7518 Bretten.

DOS-Farbgenerator

Wer stört sich nicht täglich an der vom Atari-Betriebssystem voreingestellten Hintergrundfarbe? Vom Basic oder diversen anderen Sprachen aus läßt sich die gewünschte Farbe, erfahrungsgemäß meist Schwarz, einstellen. Ein dunkler Hintergrund schont zudem die Bildröhre der Monitore (vor allem der bernsteinfarbenen). Ein Neueinstellen der Farbe ist selbst beim Drücken der RESET-Taste leicht möglich. Wer allerdings das DOS-Menü aufruft, hat Schwierigkeiten, die Farbe zu ändern. Hierfür gibt es nun eine Lösung, den "DOS-Farbgenerator".

Er arbeitet ausschließlich unter DOS 2.5. Mit diesem Programm wird direkt auf der Diskette das DUP .SYS so modifiziert, daß bei jedem Sprung in das DOS-Menü automatisch die gewünschte Schrift- und Hintergrundfarbe eingestellt wird.

Die Bedienung des Farbgenerators sieht folgendermaßen aus. Nach dem Start des Programms erscheint die Frage nach der Schriftfarbe. Hier ist eine Zahl von 0 bis 255 einzugeben, die sich aus Farbe * 16 + Helligkeit zusammensetzt. Dann ist entsprechend die Hintergrundfarbe zu wählen. Vorgegeben sind 12 für die Schrift und 0 für den Hintergrund, was weißer Schrift auf schwarzem Hintergrund entspricht. Diese Werte ergeben ein sehr angenehmes Bild. Anschließend ist nur noch eine Diskette mit DOS 2.5 einzulegen (bitte vergewissern, daß es wirklich DOS 2.5 ist, da das Programm dies nicht überprüft) und die RETURN-Taste zu betätigen. Wenn Sie nun ins DOS gehen (vorausgesetzt, Sie haben im Moment keine RAM-Disk eingeschaltet), werden Sie Ihre gewünschten Farben sehen. Außerdem ist das Menü mit einem neuen Titel versehen, da ein Teil des Originaltextes dem Unterprogramm zur Einstellung der Farben weichen mußte.

8 Bit

Wer möchte, kann auch sein eigenes Unterprogramm einbauen, in dem z.B. nur die Hintergrundfarbe eingestellt und statt der Schriftfarbe der Ladeton (POKE 65,0) ausgeschaltet wird. Dafür müßten folgende Änderungen vorgenommen werden: In Zeile 490 ist die zweite 141 gegen eine 234 zu ersetzen, in Zeile 500 müssen statt 197,2 die Zahlen 133,65 stehen. Außerdem muß die Schriftfarbe unbedingt auf 0 gestellt werden.

Burkhard Wagner

```
100 GRAPHICS 0:DIM A$(1)
110 POKE 709,14:POKE 710,0
120 POSITION 8,0
130 ? "Farbgenerator DOS 2.5"
140 POSITION 10,1
150 ? "(c) B.Wagner 8/86"
160 ? :?
170 ? "Hintergrundfarbe  0";
180 ? CHR$(30);CHR$(30);
190 INPUT H
200 ?
210 ? "Schriftfarbe 12";
220 ? CHR$(30);CHR$(30);CHR$(30);
230 INPUT S
240 ? :?
250 ? "Lege DOS 2.5 Disk ein. <RETURN>
";
260 INPUT A$
270 XIO 36,#1,0,0,"D:DUP.SYS"
280 OPEN #1,12,0,"D:DUP.SYS"
290 FOR I=1 TO 9
300 GET #1,A
310 NEXT I
320 FOR I=10 TO 48
330 READ A
340 IF I=11 THEN A=H
350 IF I=16 THEN A=S
360 PUT #1,A
370 NEXT I
380 FOR I=49 TO 431
390 GET #1,A
400 NEXT I
410 PUT #1,32
420 PUT #1,15
430 PUT #1,31
440 CLOSE #1
450 XIO 35,#1,0,0,"D:*.SYS"
460 ? :?
470 ? "O.K.":?
480 END
490 DATA 169,0,141,198,2,169,14,141
500 DATA 197,2,173,198,25,96
510 DATA 125,65,66,66,85,67,32,101
520 DATA 46,86,46,32,68,79,83,32
530 DATA 50,46,53,70,97,114,98
540 DATA 101,155
550 REM ----------------------------
560 REM - ATARI BIT BYTER USER CLUB -
570 REM -    C/O WOLFGANG BURGER    -
580 REM -      WIESCHENBECK 45      -
590 REM -        4352 HERTEN        -
600 REM -                           -
610 REM -   (C) 1986 BY B. WAGNER   -
620 REM ----------------------------
```

Das Zweitlaufwerk

Mit der Padercomp-Floppy FL-1 können Sie die Peripherie Ihres ST preiswert ergänzen.

Laut einer Umfrage im kleinen Kreis besteht die Grundausstattung der meisten ST-Benutzer aus Computer, S/W-Monitor und einem Floppy-Laufwerk. Zur Zeit besitzen nur relativ wenige den teuren, aber hervorragenden Atari Farbmonitor bzw. ein Zweitlaufwerk. Daß ein Farbmonitor, der rund 1.300 DM kostet, nicht mal eben so angeschafft werden kann, ist verständlich. Zweitlaufwerke sind aber mittlerweile erschwinglich geworden, wenn man sich einmal die Anzeigen der verschiedenen Hersteller ansieht.

Wir haben eines dieser Laufwerke getestet. Es handelt sich um das 3,5"-Gerät FL 1 der Firma Padercomp, das als Erst- oder Zweitlaufwerk eingesetzt werden kann. Mit einem Preis von 448.– DM ist es wohl für viele Anwender interessant.

Wer schon einmal einen Ordner mit 37 Dateien mit nur einem Laufwerk auf eine andere Diskette kopiert hat, wird sicher über die Anschaffung eines Zweitlaufwerks nachgedacht haben. Abgesehen von dem hohen Zeitaufwand (bedingt durch den ständigen Diskettenwechsel) trägt diese Kopiermethode weder zur Laufwerk- und Diskettenschonung noch zur Datensicherheit bei. Auch kann das permanente "Stecken Sie Disk B in Laufwerk A – Stecken Sie Disk A in Laufwerk A" usw. besonders bei längeren Kopiervorgängen zu Verwirrung und entsprechenden Fehlern führen. Außerdem gibt es mittlerweile eine Menge Programme, die für den Einsatz einer zweiten Diskettenstation wie geschaffen sind.

Das Gerät besticht schon durch seine äußere Erscheinung. Farblich genau dem Atari angeglichen, sind es die Abmessungen, die mir besonders gefallen. Mit einer Größe von nur 105 × 40 × 240 mm (B×H×T) wirkt das Laufwerk sehr zierlich, was der Leistungsfähigkeit aber keineswegs abträglich ist. Im stabilen Blechgehäuse befindet sich neben einem NEC-Industrielaufwerk auch das Netzteil. An der Rückseite sind die beiden Kabel zur Stromversorgung und zum ST herausgeführt. Das letztere hat eine Länge von rund 50 cm, was als ausreichend bezeichnet werden kann.

Nach dem Herstellen der Verbindungen ist das Laufwerk sofort einsatzbereit. Das Gerät verfügt zwar über einen eigenen Einschalter, doch wird man leider im unklaren darüber gelassen, ob es bereits eingeschaltet ist oder nicht. Eine Leuchtdiode wäre hier sehr sinnvoll. Angezeigt wird lediglich, wenn auf eine Diskette zugegriffen wird. Das ist aber auch schon der einzige Nachteil, der mir in einem mehrwöchigen Test aufgefallen ist. (Verbunden war das Laufwerk mit einem Atari 1040 STF, es arbeitet aber laut Herstellerangabe mit jedem Atari zusammen.)

Während des Tests traten keinerlei Ausfälle auf. Überrascht haben mich immer wieder die geringen Arbeitsgeräusche der Station, die eigentlich nur bei absoluter Stille zu hören sind. Das Laufwerk besitzt durch die doppelseitige Auslegung eine Kapazität von rund 726 KByte formatiert. Alle Anwendungen funktionierten einwandfrei. Vom Kopieren einzelner Files und ganzer Disketten über das Formatieren im B-Laufwerk bis zum Starten diverser Programme klappte alles vorzüglich. Das FL-1-Laufwerk kann deshalb, nicht zuletzt auch wegen der relativ geringen Anschaffungskosten, nur empfohlen werden.

Technische Daten:

Kapazität:
1 MByte unformatiert
726 KByte formatiert

Format: 3,5" doppelseitig

Spuren pro Seite: 80

Spurdichte: 135 tpi

Aufzeichnungsverfahren/-dichte: MFM/8717 bpi

Zugriffszeiten:
Spur zu Spur 3 ms
Einstellzeit 15 ms

Umdrehungsgeschwindigkeit: 300 Upm

Datenübertragungsrate: 250 KByte/sec

MTBF 12.000 POH / MTTR 0,5 Std.

Stromversorgung: 220 V

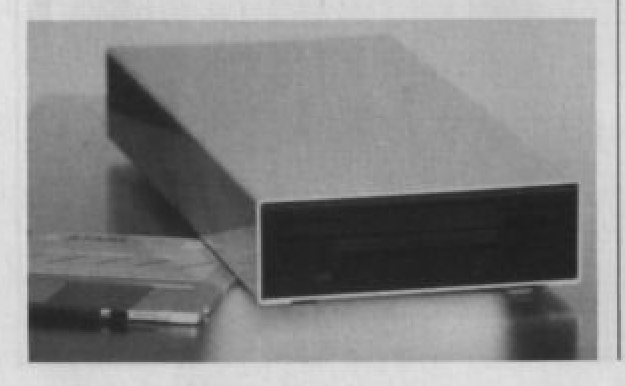
Stromverbrauch:
Standby 0,3 W / Aktiv 2,1 W

Gewicht: 1500 g

System: alle Atari ST

Hersteller/Bezugsquelle: Padercomp, Paderborn

Rolf Knorre



Peripherie im passenden Design zum ST



Sehen Sie den kleinen Mann rechts oben in der Ecke? – Nennen wir ihn einfach einmal Wilfried. Wilfried ist ein leidenschaftlicher Nasenbohrer und hat deshalb keine Lust, die Super-Listings aus dem ATARI*magazin* abzutippen. Da opfert er lieber 20 Eier und bestellt sich die Diskette zum Heft. Und wer das 1. ATARI*magazin* verpaßt hat, ist zwar selber schuld, kann aber seinen Fehler leicht ausbügeln. – Einfach diese Diskette gleich mitbestellen.

16 Bit

ATARI *magazin* 1/87

Gem-Routinen
Dateiinfo
Puzzler
Nützliche Routinen für Assemblerprogrammierung
Zusatz: 3D Flying Ace mit SW-Monitor (aus CK 11/86)
Best.-Nr. LF 16-187

ATARI *magazin* 2/87

Directory in GFA
Dateiverschlüsselung
Memorix
Programm zum Bericht "Märklin Digital"
Best.-Nr. LF 16-287



ATARI *magazin* 1/87

XL-TOS
Kreisler
Vectorgrafik in Action!
Listing zur Happy-1050-Diskettenstation
Best.-Nr. LF 8-187

ATARI *magazin* 2/87

Demo zur Zeichenanimation
Star Castle
Listing zur Serie "1050-Floppy mit Enhancement"
Test-Programm zum Bericht "320 KByte im Eigenbau"
K A H
DOS-Farbgenerator
Best.-Nr. LF 8-287

Jede Diskette kostet DM 20,-. *Bitte bestellen Sie bei* **ATARI magazin**, *Postfach 1640, 7518 Bretten. Bei Versand per Nachnahme werden Versandkosten von DM 5,70 berechnet, bei Vorauskasse keine Versandkosten (Scheck beilegen oder überweisen auf Postgirokonto Karlsruhe 434 23-756).*

Leserecke!

Wer sitzt nicht gelegentlich haareraufend vor dem Computer, und es gelingt einfach nicht, das dritte Bild zu überleben? Oder weil das Anwenderprogramm an entscheidender Stelle genau das nicht tut, was wir erwarten?

Mit Ausdauer oder Glück gelingt aber auch manche Entdeckung, die Anleitung oder Handbuch verschweigen. Damit aber nicht jeder in seinem stillen Kämmerlein das Rad neu erfinden muß, wollen wir in der Leserecke Hilfesuchende und Experten unter unseren Leser zusammenbringen.

Schildern Sie uns die Probleme mit Ihren Programmen; schreiben Sie uns Ihre Entdeckungen. Ihre Fragen und Anregungen werden im ATARI *magazin* veröffentlicht.

Spiele

Liebe Leser, willkommen bei der zweiten Spieleecke. Eigentlich sollte diesmal eine Karte zu "Alternate Reality: The City" erscheinen, doch wurde erst kürzlich in einem anderen Magazin eine sehr gute Karte veröffentlicht. Deshalb haben wir auf den Abdruck verzichtet.

An dieser Stelle möchte ich ein paar Tips zu aktuellen Adventuregames geben:

Infocoms "Leather Princesses of Phobos" bringt zur Zeit viele Köpfe zum Rauchen. Raubkopierer haben bei diesem Spiel keine Chance, denn Tips zum Durchqueren der Katakomben und Hinweise zum Entschlüsseln von Geheimbotschaften finden sich in dem beiliegenden 3-D-Comicbuch. Abenteurer, denen es im Exit-Shop an Kleingeld fenlt, sollten eine Reise zum Südpol unternehmen. Die direkte Route ist aber nicht ungefährlich. Durch Drücken zweier Knöpfe, geduldiges Warten und Betreten eines unter dem Palast versteckten schwarzen Kreises kommt man sicherer ans Ziel. Wer in Cleveland den Sturz aus dem Fenster durch die zu eifrigen Rettungskräfte nicht überlebt, sollte sich eines alten Ausbrechertricks bedienen, um sicher an den Scheinwerfer zu gelangen. Abenteurer in den Händen verrückter Wissenschaftler mit schrecklichem deutschen Akzent müssen nicht auf den Gummischlauch im Affenkäfig verzichten. Laßt Euch ruhig als Versuchskaninchen mißbrauchen, ein Doping Eurer neuen Erscheinungsform hilft Euch aus der Klemme. Wenn man das Kaninchen, das bei der Oase zu finden ist, in den T-Remover steckt, präsentiert sich ein orgineller Infocom-Scherz.

Bei Infocoms "Suspect" muß man nicht unbedingt immer als vermeintlicher Mörder abgeführt werden. Ein Verstecken der Leiche hilft nicht bei der Lösung der Falles, sondern hat nur ironische Kommentare des Computers zur Folge. Besser ist es, sich die Maske des Mordopfers genau anzusehen und daran zu denken, daß bei einem Maskenball die wahre Identität der Beteiligten schwer zu erkennen ist. Die Kostüme der Verdächtigen lassen gut auf ihre Beziehungen untereinander schließen. Ein Aufenthalt am Kamin zur rechten Zeit, das heimliche Belauschen eines Gesprächs in der Bibliothek und der Inhalt eines Kofferraums tragen zur Lösung des Falls bei. Auch darf man nicht vergessen, allen Gästen seine Version des Mordes zu erzählen und die gefundenen Indizien dem Inspektor auszuhändigen.

In "Ballyhoo" hebt Helium die Stimmung. Wer mit seiner Eintrittskarte nicht zurechtkommt, sollte sich die der Verpackung beiliegende genau anschauen. Ein geschicktes Täuschungsmanöver verkürzt die Wartezeit am Zirkuseingang im Alptraum des Hypnotiseurs.

In das Schloß des "Sorcerer of Glaymourge" führen zwei Wege. Der richtige bringt die Spielfigur ziemlich außer Atem. Das Geheimnis des Brunnens wird enthüllt durch einen Blick in den Zauberspiegel. Ein Souvenir aus dem staubigen Raum setzt den Drachen in Bewegung.

Ein Ausflug in schwindelnde Höhen läßt den Abenteurer "Mordons Quest" beginnen. Von dem mörderischen Hobbykoch befreit ein gezielter Schuß aus einem selbstgebastelten Blasrohr.

"Transylvania" bereitet in letzter Zeit hauptsächlich ST-Usern Kopfzerbrechen. Gegen den Werwolf hilft die klassische Silberkugel. Beim Hantieren mit den Jagdtrophäen findet man den Geheimraum des Magiers. Ein Schwenken des Ringes bringt außerirdische Hilfe.

Wer im "Asylum" eine Karte zeichnet, muß darauf achten, daß hier nicht alle Abzweigungen einen Winkel von 90 Grad aufweisen. Auch sollte man den cineastischen Gesichtschirurgen mehr als einmal aufsuchen.

Die Freunde von Actiongames sollen hier natürlich auch nicht zu kurz kommen. In Frank Cohens "Ghostchaser" kann man mit Hilfe des Codeworts FANDA die Hälfte des Spiels überspringen; bei "Ollies Follies" bewirken FRANK und FANDA Wunder.

In "Ninja" hilft ein kleiner Trick, die Gegner zu besiegen: Nach Verschleudern der Sterne sollte man den Raum schnell verlassen. Bei erneutem Betreten liegen alle Sterne, auch die, welche von feindlichen Ninjas aufgenommen wurden, auf dem Boden verstreut, können eingesammelt und wieder verwendet werden. Nach der Erbeutung von sechs Statuen öffnet sich ein Geheimraum, in dem sich die letzte Statue befindet. Geht man mit dieser zum Startpunkt zurück, ist das Spiel gewonnen.

Den Karateka im gleichnamigen Spiel greift der Oberschurke nur an, wenn er selbst einen Energiepunkt zurückbekommt. Man muß also versuchen, ihm mit genauen Schlägen bei jedem Angriff immer zwei Punkte abzunehmen, und das Spiel ist (fast) gewonnen.

Zum Abschluß noch ein paar News für alle Spielefreunde. Die Firma Epyx hat einen deutschen Programmierer beauftragt, ihre Produkte für den 8-Bit-Atari umzusetzen. Bleibt zu hoffen, daß bald "Winter"-, "World"- und "Summergames II" erhältlich sind. US-Gold hat die Rechte für "Questron" und "Gemstone Warrior" gekauft und wird diese Rollenspiele bald günstig auf den deutschen Markt bringen.

Bei Origin Systems/Electronic Arts tut sich auch einiges. Der Klassiker "Ultima I" wird in überarbeiteter Version wieder vertrieben. Neu im Programm sind auch "Ogre", ein futuristisches Wargame für zwei Spieler, das schon lange als Brettspiel zu haben ist, weiter "Autoduell", ein Rollenspiel für Highway-Schlachten à la "Mad Max", und "Lords of Conquest", ebenfalls ein computerisiertes Brettspiel, in dem mehrere Spieler als Herrscher verschiedener Nationen um die Vorherrschaft auf einem Kontinent kämpfen.

Tschüs bis zur nächsten Spieleecke.

Euer Frank

8 Bit

Lassen Sie sich über-rollen

16-K-Bibomon Maschinensprachemonitor mit eingebautem DOS, Hardcopy-Routine, Centronics-Port, Quarzuhr
448.- DM

Biboburner Eprommer für alle Atari-Computer, bekommt Strom vom Computer, Programm im Eprom
298.- DM

Kyan Pascal 2.0 Ein Pascal-Compiler für XL/XE-Computer, ausgelegt für die Arbeit mit einem Laufwerk. Liefert echten Object-Code
248.- DM

Kyan Pascal Toolkit 1 Das erste Toolkit für Kyan Pascal
98.- DM

Kyan Pascal Toolkit 3 Das dritte Toolkit für Kyan Pascal
98.- DM

Oldrunner + 4-K-Bibomon Die Oldrunnerplatine wird mit dem 4-K-Bibomon geliefert (Maschinensprachemonitor)
178.- DM

Oldrunner + 4 K + 80 Z. Wie oben, zusätzlich mit 80-Zeichen-Karte
198.- DM

Snaut-Forth 83 Ein Forth-83-Compiler vom Compy-Shop. Wird geliefert auf Diskette oder Cassette.
69.- DM

Speedy 1050 N Normalausführung der Speedy, macht das Laufwerk schneller, bringt eine höhere Speicherkapazität
198.- DM

Speedy 1050 T Wie oben, zusätzlich mit Trackanzeige
298.- DM

Speedy Buch Mit dem kompletten, dokumentierten ROM-Listing der Speedy 1050
50.- DM

Speichererweiterung 192 K Speichererweiterung für den Atari 600 XL von 16 K RAM auf 192 K RAM
198.- DM

Speichererweiterung 320 K Speichererweiterung für den Atari 800 XL von 64 K RAM auf 320 K RAM, mit spez. DOS
248.- DM

Speichererweiterung 64 K Speichererweiterung für den Atari 600 XL von 16 K RAM auf 64 K RAM
98.- DM

Bestellcoupon Bitte liefern Sie mir folgende Artikel:

Anzahl	Artikelbezeichnung	Preis

Ich wünsche folgende Bezahlung:
☐ Nachnahme (+ 5,70 DM Porto + Versandkosten)
☐ Vorauskasse (keine Versandkosten)

Bei Vorauskasse bitte Scheck beilegen oder auf Postgirokonto Karlsruhe 404 23-756 überweisen.

Name:
Anschrift:
PLZ/Ort:
Telefon:
Datum:
Unterschrift:

Coupon ausschneiden, auf Postkarte kleben und einsenden an:
ATARImagazin
Postfach 1640
7518 Bretten

Das Turbo-Basic für den ST

Mit dem Omikron-Basic-Interpreter in der Version 2.0 ist Basic als Programmiersprache endgültig rehabilitiert.

Bisher waren diejenigen, die ihren ST nicht mit Assembler, C oder anderen schweren Kalibern zu Leibe rükken konnten oder wollten, schwer benachteiligt. Die mühselig erworbenen Basic-Kenntnisse nutzten nicht viel, da die mitgelieferte Language-Disk eher auf die Sondermülldeponie gehört. Man erinnert sich nur ungern an die Pilze bzw. Bomben und all die anderen Feinheiten, die einem da geboten wurden. Den Eigenschaften der Maschine wird der Interpreter sowieso nicht gerecht. Viele warten wohl heute noch auf die erste funktionstüchtige Version des ST-Basic.

Doch jetzt gibt es eine mehr als vollwertige Alternative. Wen die fehlenden Zeilennummern des GfA-Basic abschreckten, der kann jetzt seinen ST mit Hilfe der neuesten Version des Omikron-Basic (Zeilennummern können ausgeblendet werden) in einen wahren Zauberkasten verwandeln. Omikron-Basic ist voll kompatibel zu M-Basic, der wohl fundamentalsten Form des Basic, die auch dem ST-Basic als Grundlage diente.

Zwei findige Programmierer haben Positives aus fernen Computerwelten zusammengetragen, speziell für den ST aufbereitet, in drei 27256-EPROMs gequetscht und aus dem Atari ST endlich einen Superrechner für den Hausgebrauch gemacht. Sogar der 260 ST dürfte damit auch ohne ROMs ein paar Byte für Basic-Programme abzweigen können. Man verfügt plötzlich über Befehle, die in ihrer Funktion dem MS-DOS entnommen wurden. So z.B.

CHDIR	Directory wechseln
MKDIR	Directory einrichten
RMDIR	Directory löschen
COPY	Datei(en) kopieren
BACKUP	.BAK-Datei erzeugen
KEY	Funktionstaste belegen
KEY LIST	Belegung anzeigen

Eigene Basic-Befehle sind in Form von Prozeduren und Funktionen definierbar. Masken lassen sich mit Hilfe des stark erweiterten INPUT-USING-Befehls derart gut gestalten, daß COBOL und dBase Mühe haben, dem Benutzer ähnlichen Komfort zu bieten.

Aber fangen wir doch vorne an. Was nützt einem der schönste Interpreter ohne Handbuch. Trotz einiger weniger Druckfehler ist das 183 Seiten starke Spiralringbuch ein gelungenes Nachschlagewerk mit echten Referenzlisten und Tabellen im Anhang. Es ist durchweg übersichtlich gestaltet und flüssig zu lesen. Spezielle Wissensbomben sind vorsorglich mit dem Vermerk "für Profis" gekennzeichnet, sollten aber die Mutigen unter uns auf keinen Fall vom Programmieren abhalten. Einmal im Basic, bewirkt selbst der Reset-Knopf keinen schwarzen Bildschirm.

Zuerst müssen wir den Interpreter installieren. Die kleine Platine wird dazu einfach vor dem Einschalten des ST in dessen Expansionsport gesteckt. Auf dem Bildschirm erscheint nach dem Booten ein neues Icon "C ROM-Modul". Durch Anmelden von Laufwerk 0 erhält man das Programm OM-BASIC.PRG. Icon angeklickt, und in Nullkommanichts sieht man die Bereitschaftsmeldung des Interpreters. Beim ST+ ohne ROMs hieß es da ganz unverhofft 656882 Bytes free – Na ja!

Kommen wir nun zum Thema "Strukturiertes Basic". Dieser Interpreter bietet genügend Möglichkeiten, übersichtliche Listings zu basteln. Da wären zunächst einmal die Labels. Diese haben die nicht zu unterschätzende Eigenschaft, mehr Aussagekraft zu besitzen als Zeilennummern. Andererseits kann das Ziel von GOSUB, GOTO und RESTORE auch weiterhin eine Zeilennummer, eine Formel oder sogar ein String sein. Mit Hilfe der Strings kann man auch Sprungziele aus einer Liste auswählen.

Prozeduren mit Namen, Rückgabeparametern und lokalen Variablen werden wohl selbst eingefleischten Pascal-Enthusiasten einen erstaunten Blick abringen. Und auch die gute, alte Funktion hat sich stark gemausert. Man ist nicht mehr auf eine Zeile angewiesen. Funktionen und Prozeduren sind rekursionsfähig und stehen, einmal definiert, auch im Direktmodus zur Verfügung.

Zur Konstruktion von Schleifen verwendet man üblicherweise REPEAT...UNTIL, WHILE ...WEND und FOR...NEXT. Allerdings verläßt man eine solche Schleife nicht mehr mit einem IF...GOTO, was ja den Interpreter bei der Schleifenverwaltung erheblich durcheinander bringen würde, sondern mit dem eigens dazu geschaffenen Befehl EXIT. Diesem kann man zusätzlich ein Ziel und die Anzahl der zu verlassenden Schleifen angeben.

Die gewohnten IF...THEN ...ELSE-Spaghettis werden jetzt

entwirrt, indem man die ganze Konstruktion über mehrere Zeilen verteilt und mit ENDIF abschließt. Das Einrücken der Zeilen wird einem zum Glück noch selbst überlassen. Mit Hilfe der TAB-Taste kann man dies ohne große Mühe bewerkstelligen, wenn man will.

Um all diese schönen Dinge überhaupt in den ST hinein zu bekommen, benutzt man am besten die Tastatur. Aber die hat es plötzlich in sich. Mit EDIT kommt man in einen richtigen Wortprozessor, der Blockoperationen, drei Schriftgrößen, Split-Screens (bis zu drei Stück, je nach Schriftgröße), eine Repeatfunktion, Suchen, Ersetzen, Help-Page und andere nützliche Funktionen beherrscht. Wenn man sich nicht im Editiermodus befindet, so kann man trotzdem Änderungen am Listing vornehmen. Es stehen dann nur nicht alle Editierfunktionen zur Verfügung. Aber immerhin wird das Listing gescrollt, wenn man den oberen oder unteren Rand des Bildschirms mit dem Cursor überschreiten würde.

Mit dem Befehl FILES wird das Directory angezeigt. Aber auch hier gibt es etwas Neues. Vor jedem File mit der Extension ".BAS" steht LOAD und vor Subdirectories FILES. Zusätzlich befindet sich hinter dem Filenamen als REM die Filelänge. Um eine Datei zu laden, fährt man den Cursor mit den Pfeiltasten in die gewünschte Zeile und drückt auf die ENTER-Taste.

Bei der Einführung in die Arithmetik kommen wir gleich zur Genauigkeit, damit ja kein Aha-Feeling aufkommt. Das Problem 1/3 ist wegen seiner Unendlichkeit immer noch nicht ganz gelöst. Ansonsten dürfte die 19stellige Genauigkeit bei jedem, der sich gestern noch mit Systemabstürzen des ST-Basic bei größeren doppeltgenauen Fließkommaoperationen herumgeärgert hat, für inneren Frieden sorgen. Etwas eigenwillig und gewöhnungsbedürftig ist dagegen der Umgang mit Byte-, Word- oder Longintegers. Der Wert −10 wird nicht etwa als FFF6 dargestellt, sondern eben als −A. Den üblichen Wert −1 für FFFF erhält man nun durch LOW ($FFFF). HEX $(HIGH ($7FFFFFFF)) ergibt 7FFF. Alle Variablen sind Longintegers, solange sie nicht anderweitig gekennzeichnet oder global umdefiniert wurden. Dem entsprechend gibt es auch drei Formen von PEEK und POKE.

Damit Sie gleich sehen, was mit Omikron-Basic alles möglich ist, anschließend ein paar Tabellen.

Variablentypen

Variable%F	Flag
Variable%B	Integer-Byte
Variable%W	Integer-Word
Variable%L	Integer-Long
Variable!	Float-Single
Variable#	Float-Double
Variable$	String

Logische Operatoren

NOT	Nicht
AND	Und
OR	Oder
XOR	Exklusiv-Oder
EQV	Äquivalent
IMP	Implikation
NAND	Nicht-Und
NOR	Nicht-Oder
SHR	Schiebe rechts
SHL	Schiebe links

Mathematische Funktionen sind in einer solchen Anzahl vorhanden, daß man die Qual der Wahl hat.

Zusätzliche mathematische Funktionen

FRAC	Nachkommastellen
DEG	Gradmaß
RAD	Bogenmaß
SEC	Sekans
COSEC	Cosekans
ARCTAN	Umkehrfunktionen
ARCSIN	
ARCCOS	
ARCCOT	
SINH	Hyperbolische
COSH	Winkelfunktionen
TANH	
COTH	
SECH	Secans hyperbolicus
COSECH	
ARSINH	areasinus
ARTANH	hyperbolicus
ARCOTH	
LOG	Logarithmus von C zu B A = LOG (B,C)
FACT	Fakultät
BIT	Testet Bit B von C A = BIT (B,C)

Sonstige neue Funktionen

MIN	Den kleineren zweier Werte auswählen
MAX	Den größeren zweier Werte auswählen
FRE	a) freien Speicher ermitteln b) Garbage Collection c) freien Diskettenspeicher ermitteln

Zahlensysteme

HEX$	Hexzahl ermitteln
OCT$	Oktalzahl ermitteln
BIN$	Binärzahl ermitteln
&H	Hexzahl
&D	Dezimalzahl
&O	Oktalzahl
&B	Binärzahl
$FFFF	Hexzahl
65535	Dezimalzahl
&177777	Oktalzahl
%111...	Binärzahl

Etwas Inkompatibilität zum M-Basic besteht allerdings. Die Befehle MKI$, MKS$, MKD$ sowie die Funktionen CVI, CVS und CVD berechnen aus übertragenen Dateien andere Werte. Dies läßt sich dadurch umgehen, daß man diese Werte im bisherigen Basic mit STR$ umwandelt und abspeichert, bevor man sie mit Omikron-Basic wieder ein-



Fertig für DESKTO.INF

liest. Zudem verfügt man dann auch noch über MKIL$ und CVIL.

Strings sind multiplizierbar. "A"*5 ergibt "AAAAA". Durch den Einsatz von Klammern lassen sich komplizierte Gebilde durch kurze Formeln erzeugen. Die gängigen Stringfunktionen wie MID, LEN etc. wurden um MIRROR, UPPER und LOWER erweitert. Abfragen wie IF a$="j" OR a$="J" gehören damit der Vergangenheit an.

Für professionelle Mathematiker sind Funktionen zur Matrizenalgebra installiert. Möglich sind die Erzeugung einer Einheitsmatrix, das Multiplizieren eines Arrays mit einer Konstanten, die Multiplikation zweier Matrizen sowie Löschen, Invertieren und Berechnen der Determinanten einer Matrix.

Mit MODE kann man zwischen drei Zeichensätzen wählen, und zwar D, USA und GB. Als Systemvariablen stehen die seit dem Einschalten des STs verflossene Zeit, die aktuelle Uhrzeit, das aktuelle Datum, Pi, die aktuelle Cursorzeile und die Fehlervariablen ERR, ERL und ERR$ zur Verfügung. Auch die XY-Koordinaten der Maus sowie der Status der Mausschalter sind leicht zu erfragen. Die Maus kann zudem an- und abgeschaltet werden.

Die USING-Funktion wurde etwas erweitert und kann auch global eingesetzt werden. Dann gilt sie sogar für die STR$(x)-Funktion. Um den Cursor zu positionieren, genügt PRINT @ (Zeile, Spalte). Positionen können auch mit x$=@ (Z,S) festgehalten werden.

Eine Besonderheit bei Omikron: der Interpreter im EPROM.



Bei den INPUT-Befehlen wird man stark an dBase erinnert. Auch läßt sich die @ (z,s)-Funktion einbauen. Aber dazu kommt noch ein Steuerstring, mit dem man die Eingabe vielfach eingrenzen kann. Auch die Festlegung der maximalen Eingabelänge ist möglich. Das Füllzeichen für unbenutzte Stellen ist frei definierbar. Sollte in der Eingabevariablen schon ein String stehen, so wird dieser dargestellt und kann dann geändert werden. Hier kann auch die Cursorposition innerhalb dieses Strings angegeben sein. Zudem läßt sich die maximale Anzahl einzulesender Zeichen bestimmen. Tasten, mit denen man die Eingabe verlassen kann, müssen angegeben sein. Die tatsächlich gedrückte wird dann in einer Variablen zurückgemeldet.

Die Funktion INKEY $ liefert dadurch noch mehr Informationen als jemals zuvor. Vier Byte werden zurückgemeldet. In Byte eins ist der Zustand der beiden Shift-Tasten sowie von Control, Alternate und Capslock enthalten. Byte zwei enthält den Scancode der Taste; der ASCII-Code steht im vierten Byte.

Die zusätzlichen Funktionen der normalen Basic-Kommandos sind schon fantastisch. Aber es gibt den neuen Befehl SORT, der zusätzlich Komfort eröffnet. Normalerweise kommt bei ihm der Quicksort-Algorithmus zur Anwendung. Sollte der freie Speicher dafür nicht ausreichen, wird auf das Bubble-Sort-Verfahren umgeschaltet. Dieses Sortierverfahren ist etwas langsamer, dafür aber weniger raumgreifend. Auch im maschinennahen Bereich ist so manches möglich. Mit SEGPTR verrät Omikron-Basic die Adresse einer Adreßtabelle, die im Handbuch aufgeschlüsselt wird und für Basic-interne Variablen zuständig ist. Mit MEMORY läßt sich der noch freie Speicher des GEMDOS-Bereichs feststellen oder ändern. Zum Aufruf von Systemroutinen stehen die Befehle BIOS, XBIOS, GEMDOS, VDI und AES bereit. Aber auch eigene Maschinenprogramme sind mit CALL oder USR ansprechbar.

Um von einem Programm aus jederzeit ein anderes Programm aufrufen zu können, gibt es mehrere Möglichkeiten:

ON MOUSEBUT GOSUB
ON KEY GOSUB
ON HELP GOSUB
ON TIMER GOSUB

Dadurch werden also keine Unterprogramme beim Erreichen eines GOSUB-Befehls aufgerufen, sondern immer dann, wenn eine bestimmte Bedingung erfüllt ist. Mit der vierten Funktion kann man zum Beispiel im einfachsten Fall eine Uhr darstellen. Aber auch Verwandte von Sideklick sind vorstellbar.

```
ESC  /  UNDO      RESTORE LINE / UNDO
SF1 (=SHIFT-F1)   SPLIT SCREEN
SF2      SF3      CHANGE SCREEN / MODE
F1 n   F1 +/-n    GO TO LINE
F1 CRSR UP/DOWN   SCROLL UP/DOWN
F2-F2    F2-F3    SEARCH / LIST
F3-F2    F3-F3    QUERY REPLACE / R. ALL
F4, F5, SF4, SF5  USER-DEFINABLE KEYS
... F6 n Return   REPEAT n-1 TIMES
SF6-F9   SF6-SF9  CUT LINE / LINK LINE
F7-F7    F7-SF7   BLOCK START / END
F7-F8    F7-SF8   SAVE / LOAD BLOCK
F7-F9    F7-SF9   INSERT / DELETE BLOCK
SF7-F4 ... F4     DEFINE FUNCTION KEY F4
F8       SF8      SAVE / LOAD FILE
F9       SF9      INSERT / DELETE LINE
F10      SF10     FAST FORWARD / BACK
ALTERNATE         STOP REPEAT & SEARCH
CONTROL-DEL       ERASE END OF LINE
CONTROL-INS       INSERT-MODE
CONTROL-<, >      TO START / END OF LINE
CONTROL-S         RENUMBER BY 10
CONTROL-CLR/HOME  LINE NUMBERS ON/OFF
CONTROL-C         QUIT EDITOR
        PRESS <CR> ...
```

Die Editor-Help-Page

Zum Programmieren des Sound-Prozessors stehen drei Befehle bereit. Frequenz, Lautstärke, Hüllkurve und Rauschfrequenz lassen sich so leicht einstellen und verändern. Zusätzlich gibt es einen Befehl zur Darstellung einer File-Select-Box. Auch Alertboxen lassen sich spielend leicht verwirklichen. Zum Beispiel kann man sich damit eine schöne Fehlerabfangroutine konstruieren.

Auch die Grafik kommt nicht

zu kurz. Zwei Hardcopy-Befehle stehen bereit. Einer entspricht der ALT-HELP-Funktion, der andere ergibt eine Texthardcopy. Es können bis zu drei Screens definiert werden. Die sogenannten "Drawing Primitives" des GEM-VDI sind als Basic-Befehle frei verfügbar. Kein umständliches Werkeln mehr mit einem GEMSYS-Befehl. Zur Konstruktion und zur Darstellung von Sprites sind zusätzlich zu BITBLT noch zwei Befehle installiert.

Hier eine Liste aller Grafikbefehle

SCREEN	einen von 3 Bildschirmen anwählen, definieren
DRAW	Linie(n) ziehen
MODE	Replace, transparent, XOR, Reverse Transparent
LINE COLOR	Farbnummer
PALETTE	Farbtönung
LINE STYLE	Linientyp
FILL STYLE	Füllmuster
FILL COLOR	Farbnummer
TEXT STYLE	Buchstabenhöhe
TEXT COLOR	Farbnummer
BOX	Rechteck
PBOX	ausgefüllt
RBOX	abgerundet
PRBOX	ausgefüllt, abgerundet
CIRCLE	Kreis
PCIRCLE	ausgefüllt
ELLIPSE	Ellipse
PELLIPSE	ausgefüllt
FILL	Fläche füllen
TEXT	Text an Grafikposition ausgeben
CLIP	Grafikfenster festlegen
BITBLT	Bit-Block-Transfer
DEF SPRITE	Sprite definieren
SPRITE	Sprite darstellen
POINT	Pixel testen

Jetzt zur Fehlersuche. Jedes Programm benötigt seine Entwicklungszeit. Bis dahin muß so mancher Fehler gefunden werden. Um einen bestimmten Programmteil zu durchforsten, setzt man am besten den Befehl

```
***     OMIKRON BASIC V2.0
656882 bytes free.
OK
files
FILES"DIV_DEMO\*.*"
FILES"RCS\*.*"
'       OM-BAS-R.BAT   '9
LOAD "GEMDEMO.BAS"     '37152
LOAD "GEMLIB.BAS"      '32687
'       GEMLIB.DOC     '27679
LOAD "IS_DEMO.BAS"     '9488
LOAD "IS_LIB.BAS"      '5872
'       IS_LIB.DOC     '3122
LOAD "MK_DESK.BAS"     '1858
LOAD "MUEHLE.BAS"      '33522
'       OM-BAS-R.DOC   '1593
'       OM-BAS-R.PRG   '57737
'       OM-BASIC.PRG   '78
LOAD "PROTECT.BAS"     '2884
LOAD "SELECT.BAS"      '1224
Break
OK
█
```

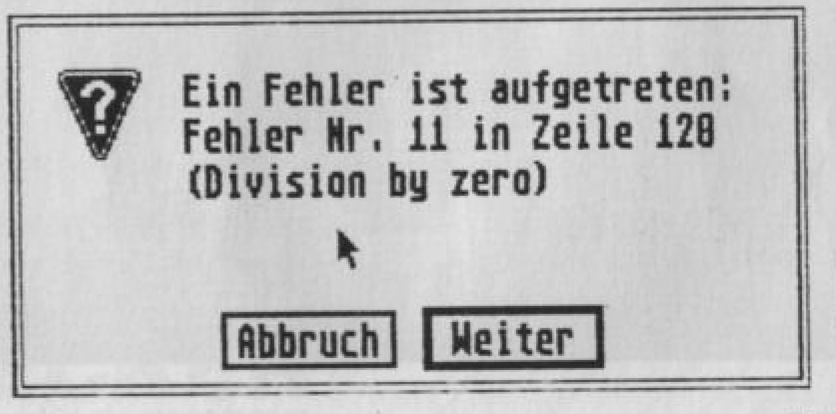
Startmeldung und der Befehl FILES

"CONT Beginn TO Ziel" ein. Mit TRON schaltet man die Zeilenverfolgung ein. Dabei wird immer der Beginn der momentan abzuarbeitenden Zeile ausgegeben. TRON kann auch mit ON TRON GOSUB zu einer eigenen Routine umgebogen werden. Zudem ist das Auflisten aller benutzten Variablen inklusive ihrer momentanen Werte mit DUMP möglich (LDUMP für den Drukker). Unter Zuhilfenahme des starken CONT-Befehls dürfte dies eine bessere Lösung als FOLLOW sein.

Ein passender Compiler soll bei der diesjährigen Hannover-Messe vorgestellt werden. Ein vollständig in Omikron-Basic geschriebenes Zeichenprogramm steht inklusive Quell-Code ebenfalls zur Verfügung. Die Besitzer einer 1.X-Version können dies gegen eine Bearbeitungsgebühr erhalten. Auf der mitgelieferten Diskette befindet sich außer einigen teilweise umfangreichen Demos ein Runtime-Interpreter, der frei kopierbar ist und mit unter Omikron-Basic erstellten Programmen verkauft werden darf, sofern es sich dabei um keine Programmiersprache handelt. Als weitere Erleichterung der Programmierung lassen sich die beiden Modul-Bibliotheken verwenden. Da wäre einmal eine vollständige GEM-Bibliothek, des weiteren eine sogenannte ISAM-Bibliothek, die umfangreiche Funktionen für indizierte Dateiverarbeitungen bietet. Diese beiden Modulsammlungen dürfen in dieser Form nicht veräußert werden, geben aber tieferen Einblick in die Handhabung des Betriebssystems. Eigene Routinen sind dann schnell verfaßt.

Fazit: Dieser Interpreter ist mehr als brauchbar! Wer ihn erst einmal hat, möchte ihn nicht mehr missen. Sicher wird es noch zukünftige Versionen geben, auf die man aber bestimmt nicht warten sollte, denn die vorliegende Version 2.0 läßt sich nicht mehr als Vorabversion bezeichnen. Das ST-Basic wird es sehr schwer haben, diesen Entwicklungsvorsprung einzuholen. Beim Marketing hat Omikron gegenüber GFA noch einiges nachzuholen. Das ist aber nicht Sache der Programmierer. Die haben den Turbo für das Aufholmanöver bereits konstruiert.

H. H. Fischer



Mit dem Basic-Befehl FORM-ALERT erzeugen Sie solche Windows.



Videobilder digital

Vielleicht muß man überhaupt nicht mehr erklären, worum es sich bei einem Digitizer handelt bzw. was man mit einem solchen Gerät anfangen kann. Wer sich schon länger mit Computern beschäftigt, wird sicher schon davon gehört haben. Da sich in der Vorweihnachtszeit gerade die kleinen Ataris sehr gut verkauft haben sollen, gibt es aber sicher auch genug Einsteiger, für die ich das Prinzip kurz erläutern möchte. Das ist schon mit wenigen Worten möglich.

Im Grunde sorgt ein Digitizer dafür, daß mit einer Videokamera aufgenommene Bilder für die weitere Bearbeitung und Verfremdung mit einem Computer entsprechend umgesetzt werden. Anders ausgedrückt ist dieses Gerät die Schnittstelle zwischen Videokamera oder -recorder und dem Computer. Die besten Ergebnisse lassen sich übrigens erzielen, wenn ein Motiv mit einer Videokamera via Videorecorder in den Digitizer eingespeist wird. Natürlich kann man auch nur den Recorder verwenden, muß dann aber ein sehr ruhiges Standbild haben.

Eigene Bilder im Computer: Mit dem Videomeister keine Kunst.

Einen solchen Digitizer bietet jetzt die Firma Irata für die 8-Bit-Ataris an. Damit wird meiner Meinung nach eine echte Marktlücke geschlossen, da es solche Geräte für andere Computertypen schon lange gibt. Bemerkenswert ist auch der Preis von 199.– DM (inkl. Software). Für den ST bekommt man einen Digitizer z.B. erst für rund 500 DM.

Uns stand zum Test ein Vorabexemplar zur Verfügung, das noch nicht richtig beschriftet war. Auch lag ihm keine besondere Anleitung bei. Diese ist aber auch nicht unbedingt notwendig, da die Anwendung des Gerätes recht einfach ist. Verbunden wird der Digitizer mit beiden Joystick-Ports und der Bildquelle, also einer Kamera oder einem Videorecorder (ein entsprechendes Verbindungskabel wird leider nicht mitgeliefert). Auf der Oberseite des Digitizers befinden sich 5 Regler, die folgende Aufgaben haben:

- Einstellung der Bildbreite
- Einstellung von Synchron und Kontrast
- Feinabstimmung

Hat man die entsprechenden Verbindungen realisiert und die Einstellungen vorgenommen, steht der Arbeit mit dem Videomeister nichts mehr im Wege, abgesehen vom Laden der Steuer-Software. Diese wird auf Diskette mitgeliefert und umfaßt eigentlich drei Programme. Gestartet wird jedoch immer mit dem Programmteil "Videomeister", der dem eigentlichen Digitalisierungsvorgang dient.

Wie man der Tabelle im Kasten entnehmen kann, wird mit dem Programmteil "Pic Mix" das zuvor digitalisierte Bild bearbeitet und über die Druckeroption zu Papier gebracht. Im Test traten keinerlei Probleme auf. Sowohl das Erstellen neuer Bilder als auch die Bearbeitung der abgespeicherten Demos funktionierte sehr gut. Der Digitizer Videomeister bietet damit ein hervorragendes Preis/Leistungs-Verhältnis und kann uneingeschränkt empfohlen werden.

System: Atari XL/XE
Hersteller/Bezugsquelle: Irata
Preis: 199.– DM

Rolf Knorre



Videomeister Auswahl

JUSTIEREN	Bild fangen muß SYNCHRON stehen
KONTRAST	Hell/Dunkel-Einstellung – oft verschieden
SCHWARZ-WEISS	Normale Aufnahme ohne Feinheiten
FOTO	Bild sieht aus wie ein Foto
4 PUNKT	Bild ist aufgerastert
8 PUNKT	Bild ist schärfer gerastert
HOHE AUFLÖSUNG	Bild hat auch Feinheiten (lichtabhängig)
NIEDRIGE AUFLÖSUNG	Bild ist etwas abgeschwächt (lichtabhängig)
BILD ZEIGEN	Aufgenommenes Bild auf Bildschirm bringen
BILD SPEICHERN	Bild auf Diskette speichern
DISK-INHALT	Sehen, was auf der Diskette gespeichert ist

Pic Mix

B	BILD 1 WÄHLEN	Erste Hälfte vom Bild wählen
B	BILD 2 WÄHLEN	Zweite Hälfte vom Bild wählen
L	BILD 1 LADEN	Ersten Teil laden
L	BILD 2 LADEN	Zweiten Teil laden
N	NEUSTART	Will man neu anfangen auszusuchen
Z	BILD ZEIGEN	Bild auf dem Bildschirm ansehen
T	FARBPUNKTE TAUSCHEN	Immer RETURN drücken, wenn fertig
M	E EFFECTE	Bild wird verändert
M	D DUNKEL	Bild wird dunkel gemacht (stufenlos)
M	H HELL	Bild wird heller gemacht (stufenlos)
P	POSITIV-NEGATIV	Bild wird jeweils umgekehrt
F	EINFÄRBEN	Bild wird coloriert
D	FARBBILD SPEICHERN	Extender soll .COL sein
I	FARBBILD LADEN	Nur mit Extender .COL laden

Drucker

GRAFIK 9 FOTO	Ausdruck erfolgt mit einer Sonderroutine
GRAFIK 8 S/W	Ausdruck erfolgt mit normaler Routine
BILD LADEN	Bild wird zum Drucken geladen
BILD DRUCKEN	Bild wird gedruckt
BILD ZEIGEN	Bild wird auf Bildschirm gegeben

Nachtrag zum XL-TOS

Zu dem im letzten Heft veröffentlichten Programm XL-TOS gibt es noch einen Nachtrag. Einige Programme lassen sich damit nicht laden. Diese stürzen ab, wenn die Laderoutine (etwa ab $2000) überschrieben wird. Deshalb enthält XL-TOS eine eigene Laderoutine, da sich einige Programme mit DOS 2.5 von Atari nicht laden lassen, sehr wohl aber mit XL-TOS. Außerdem kann XL-TOS auch in Verbindung mit dem OS/A+ oder z.B. MYDOS verwendet werden. Die Laderoutine funktioniert bei allen DOS-Arten, deren Filestruktur DOS 2.0 entspricht. Möchten Sie die Original-Laderoutine von DOS 2.5 verwenden, müssen folgende Programmzeilen geändert werden:

```
1030 DATA 32, 82, 39, 240, 1, 96, 169, 1, 32, 183, 36, 32, 177, 33,
     234, 234, 1855
1050 DATA 160, 4, 32, 60, 33, 76, 200, 21, 173, 117, 32, 141, 111,
     32, 173, 118, 32, 1515
```

Doch leider ist auch DOS 2.5 nicht unfehlbar. Alle Programme lassen sich trotz dieser Methode nicht laden. Des weiteren kann XL-TOS nur in Verbindung mit DOS 2.0 oder DOS 2.5 verwendet werden. Bei den anderen funktioniert die Laderoutine nicht.

Marc Ebner

Profitieren Sie von unserem Know-How!

OMIKRON-BASIC



ARITHMETIK
- Rechengenauigkeit bis 19 Stellen bei allen Funktionen
- Rechenbereich bis 5.11 E±4931

GEM
- Komplette GEM-Library
- sämtliche AES- & VDI-Funktionen direkt mit Namen verfügbar
- eigene BITBLIT-Routine
- NEU: jetzt mit Resource Construction Set

EXTRAS
- Masken-INPUT
- SORT-Befehl sortiert beliebige Felder
- Matrizenbefehle
- Compiler ab Frühjahr verfügbar

STRUKTUR
- Prozeduren und mehrzeilige Funktionen
- mit Übergabe- und Rückgabe-Parametern und lokalen Variablen
- REPEAT...UNTIL, WHILE...WEND, mehrzeiliges IF...THEN...ELSE...ENDIF
- Labels bei GOTO, GOSUB, ON...GOTO etc.

KOMPATIBILITÄT
- 99 % MBASIC-kompatibel
- Editor findet Inkompatibilitäten
- dadurch einfachste Anpassung

Editor
- Mit oder ohne Zeilennummern (umschaltbar)
- Drei Schriftgrößen bis 57x128
- Frei definierbare Funktionstasten

GESCHWINDIGKEIT
- FIT-Code (FIT = Fast Interpreting Technique)
- Wir kennen keine schnelleren 68000-Fließkommaroutinen
- Volle Integer-Arithmetik
- eigene Disk-Routinen für beschleunigten Dateizugriff

LIEFERUMFANG
- Modul mit OMIKRON-BASIC (wird seitlich eingesteckt)
- Demodiskette mit Runtime-Interpreter, Hilfsprogrammen und vielen Beispielprogrammen
- 180-seitiges, spiralgebundenes deutsches Handbuch
- Alles zus. nur DM 229,– (unverbindl. Preisempfehlung)
  Handbuch vorab DM 30,–

„Atemberaubende Geschwindigkeit" (DATA WELT 12/86, S. 69)
„Für ernsthafte Programmierer, die effektive und vor allem schnelle Programme erstellen wollen, ..., ist dieser neue BASIC-Interpreter genau das richtige" (DATA WELT 12/86, S. 70)
„OMIKRON-BASIC kann wesentlich mehr als der BASIC-Standard" (ST Computer 12/86, S. 79)

OMIKRON **Software · Erlachstr. 15 · 7534 Birkenfeld · Tel. (0 70 82) 53 86**

Österreich: Ueberreuter Media, Alser Str. 24, A-1091 Wien
Schweiz: Microtron, Postfach 40, CH-4542 Pieterlen

GFA-Painter

Von Rainer Lüers
Verlag Data Becker
381 Seiten, 39.– DM
ISBN 3-89011-249-8

Um Irrtümern gleich vorzubeugen: Die Firma GFA hat bislang kein Zeichenprogramm namens "GFA-Painter" auf den Markt gebracht und wird dies wohl auch nicht tun. Das könnte man nämlich annehmen, sieht man nur auf den Titel dieses Buches. Richtig ist, daß hier lediglich ein Listing abgedruckt wurde, das nach dem Eintippen zu "GFA-Painter" wird. Ein solches Konzept ist auf dem Buchmarkt bisher wohl einmalig. Die Bezeichnung GFA wurde dem Programmtitel vorangestellt, weil das Listing in GFA-Basic geschrieben wurde.

Das Buch umfaßt 381 Seiten, von denen rund 140 auf das Listing entfallen. Wer soll diese Datenmenge aber eintippen? Das wird man sich wohl auch beim Verlag gedacht haben, da dort eine Diskette zum Buch angeboten wird, auf der sich die lauffähige Version befindet.



Die restlichen Seiten stellen die Bedienungsanleitung des Programms dar. Man muß aber ganz klar sagen, daß sich dieses Buch nicht als Lehrbuch für GFA-Basic eignet. Dazu hätte man den Programmaufbau erheblich besser dokumentieren müssen. Als Anleitung zum Diskettenprogramm ist der Band wiederum etwas zu umfangreich und als Paket mit 68.– DM so teuer wie andere Grafikprogramme (z.B. "Jack Paint").

Bleibt also nur noch die Gruppe der ST-Besitzer, die das Listing tatsächlich eintippen. Für sie stellt das Buch eine lohnende Anschaffung dar, da es bisher noch kein Grafikprogramm zu diesem Preis gibt. Wenn man sieht, was "Monostar" leistet (weitgehend in GFA-Basic geschrieben), weiß man, was mit diesem Basic-Interpreter alles möglich ist.

Dennoch bleibt die Frage offen, wer sich der Mühe des Eintippens unterzieht. Schon bei Listings von nur 2 bis 3 Seiten Länge schleichen sich recht schnell Fehler ein – und dann rund 140 Seiten!

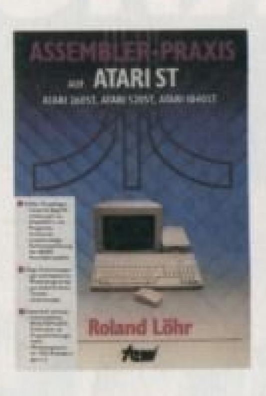
Stephan König

Assembler-Praxis auf Atari ST

Von Roland Löhr
Verlag tewi
260 Seiten, 59.– DM
ISBN 3-921803-70-5

Der Atari ST ist ein Rechner mit beeindruckenden Leistungen, doch nur ein kleiner Teil davon kann über Basic genutzt werden. Die C-Programmierer haben es da schon etwas besser; an vielen Stellen hapert es allerdings noch mit der Geschwindigkeit. Die einzige Alternative für schnelle und gute Programme lautet folglich Assembler, aber so einfach wie in den sogenannten höheren Sprachen hat man es hier nicht. Assembler-Programme werden schnell unübersichtlich, sind relativ lang und erzwingen eine tagelange Fehlersuche. Die damit verbundenen Mühen sollte man jedoch nicht scheuen. Das erste Hindernis auf dem Weg zur guten Assembler-Software ist das Erlernen dieser Sprache. Dazu sind weitgehendes Verständnis der Vorgänge im Computer auf unterster Ebene und viel Praxis notwendig.

Das vorliegende Buch verspricht nun das praktische Erlernen dieser Sprache – und es verspricht wirklich nicht zuviel. Nach Art eines Kurses entwik-



kelt der Autor eine große Anzahl von Musterprogrammen, die den Leser auch in Zukunft begleiten werden. So beschäftigt er sich mit der Textbearbeitung, dem dezimalen und binären Rechnen, der seriellen Datenübertragung und Floppy-Funktionen. Durch die gut dokumentierten Listings erlernt der Leser das Programmieren und kann sich später auf die Erstellung eigener Werke konzentrieren. Zunächst muß man sich allerdings mit der Theorie befassen. Die Struktur des 68000-Prozessors sowie die Grundbegriffe der Assembler-Sprache werden erklärt. Die verschiedenen Adressierungsarten und vor allem der Befehlsvorrat sind Dinge, ohne deren Wissen sich keine Programme erstellen lassen. Der Autor verpackt das Ganze in einen klaren Stil, der weniger attraktiv als informativ ist. Das weist eigentlich schon auf die Zielgruppe hin, erfahrene Computerfreaks, die das Letzte aus dem Atari ST herausholen wollen. Es wird doch schon eine Reihe von Kenntnissen vorausgesetzt, die man sich erst nach einiger Zeit aneignen kann.

Das Buch enthält das gesamte Wissen, um den 68000 zu programmieren und dabei die Fähigkeiten des Computers voll auszunutzen. Um dies alles verstehen und anwenden zu können, genügt allerdings die Lektüre des Buches alleine nicht. Man sollte schon ein gewisses Verständnis für die Vorgänge im Atari ST mitbringen, und vor allem sollten die Programmierkenntnisse über Basic hinausgehen. Für einen derart gewappneten Leser stellt das Buch dann eine hervorragende Informationsquelle dar.

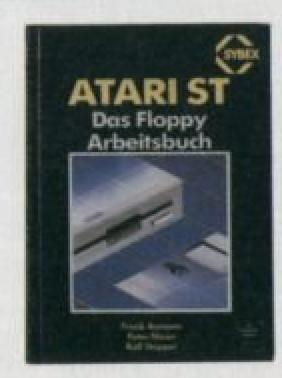
Thomas Tai

Atari ST – Das Floppy Arbeitsbuch

Von Aumann/Maier/Stöpper
Sybex-Verlag
220 Seiten, 69.– DM
ISBN 3-88745-642-4

Gleich drei Autoren haben am vorliegenden Buch mitgearbeitet, das jetzt neu auf dem Markt erschienen ist. Um es gleich vorwegzunehmen, ich halte es für sehr gelungen. Wer mit seinem Diskettenlaufwerk nicht nur Programme laden und abspeichern, sondern genau wissen möchte, was da eigentlich abläuft, sollte sich diesen Band unbedingt zulegen.

Kapitel 1 beschäftigt sich mit grundlegenden Dingen wie den verschiedenen Aufzeichnungsverfahren, dem Aufbau einer Diskette mit Spuren, Tracks und Sektoren, gilt aber auch zahlreiche Informationen darüber, was auf einer Diskette wo steht. Da geht es dann um den Boot-Sektor, das Inhaltsverzeichnis und vieles mehr. Die Materie ist zwar nicht immer leicht verständlich, aber auch für Anfänger zu beherrschen, sofern sie sich intensiv damit auseinandersetzen.



Im nächsten Kapitel wird die Diskettenprogrammierung unter TOS erläutert. TOS, GEMDOS, BIOS und XBIOS sind die Stichworte, die Einsteigern das Gruseln lehren, schon nach kurzer Einarbeitungszeit aber

viel von ihrem Schrecken verlieren. Die einzelnen Kommandos werden mehr oder weniger ausführlich erläutert. In Verbindung mit einigen Programmbeispielen kann der Leser nachvollziehen, worum es sich eigentlich handelt.

Weiter geht es mit einer Kurzbeschreibung der Diskettenschnittstelle, die sich auf die Anschlußbelegung beschränkt.

Die Direktprogrammierung des Floppy-Disc-Controllers (FDC) und sein Aufbau dürfen in einem solchen Arbeitsbuch natürlich auch nicht fehlen. In Kapitel 4 wird dieses Thema dann in gewohnter Qualität behandelt.

Im Kaufpreis ist eine Diskette mit allen Beispielprogrammen des Buches enthalten. Auf ihr befinden sich noch weitere Programme, die im Kapitel über die Powerdisk näher erläutert werden. Es handelt sich dabei um solche, die zur Arbeit mit einer Diskettenstation benötigt werden. Dort findet man ein File- und Diskettenkopierprogramm ebenso vor wie einen Diskmonitor.

Zahlreiche Tabellen im Anhang komplettieren das Buch, das ich, wie bereits gesagt, für sehr gut und fast unentbehrlich halte. Lediglich der Preis ist etwas hoch.

Rolf Knorre

Das Basic-Trainingsbuch zu Atari 600 XL/800 XL

Von Werner Voß
Verlag Data Becker
380 Seiten, 39.– DM
ISBN 3-89011-057-6

Das vorliegende Buch ist zwar nicht neu auf dem Markt, aber dennoch eine Erwähnung wert. Gerade mit den 8-Bit-Ataris hat es der Einsteiger bekanntlich nicht leicht. Die Computer bieten dem Programmierer enorme Möglichkeiten, die Anwendung ist aber selbst unter der relativ leicht zu erlernenden Sprache Basic keineswegs leicht. Aus diesem Grund gibt es mittlerweile zahlreiche Lehrbücher, die sich insbesondere mit dieser Programmiersprache beschäftigen.



Das bereits 1984 erschienene Trainingsbuch gehört zu den besseren Werken, da der Autor sehr detailliert und allgemein verständlich an die Materie heranführt. Wer den Band gründlich durchgearbeitet und alle Übungsaufgaben richtig gelöst hat (die Auflösungen sind natürlich auch enthalten), kann sich mit Recht als Basic-Programmierer bezeichnen. Die Realisation eigener Ideen dürfte danach kein Problem mehr sein.

Rolf Knorre

Das Atari-Spielebuch für 600 XL/800 XL

Von James/Gee/Ewbank
Vogel Buchverlag
184 Seiten, 30.– DM
ISBN 3-8023-0788-7

Beim vorliegenden Buch handelt es sich um eine Übersetzung des englischen Originals "The Atari Book of Games". Die Autoren bieten mit 21 abgedruckten Listings eine bunte Mischung aus bewegten Action-Spielen, aber auch ruhigeren Brett- und Textspielen. Die Zusammenstellung könnte man fast klassisch nennen, da sie so oder in ähnlicher Form schon für die verschiedensten Computer auf den Markt gekommen ist – wenn auch nicht von den gleichen Autoren.

Bei den einzelnen Spielen, auf die ich nicht näher eingehen möchte, handelt es sich durchweg um Basic-Programme ohne besonderen Anspruch. Das soll aber keine Abwertung sein, da man mit ihnen durchaus seine Freude haben kann.

Fortgeschrittene Anwender werden von diesem Band eher enttäuscht sein; für den Anfänger kann er aber den Grundstock einer eigenen Programmbibliothek bilden. Da die Listings nur mäßig dokumentiert sind, eignet sich das Buch auch nicht zum Basic-Studium.

Stephan König

Utilities in Basic für Atari-Computer

Von Alfred Görgens
Vogel Buchverlag
113 Seiten, 25.– DM
ISBN 3-8023-0854-9

Wieder eine neue Programmsammlung, war mein erster Gedanke, davon gibt es wirklich schon genug. Doch dieser erste, leicht negative Eindruck war falsch. Zwar handelt es sich wieder um eine reine Ansammlung verschiedener Basic-Programme, die aber im Vergleich mit anderen Werken erheblich besser abschneidet.

Das liegt zum Teil daran, daß sich der Autor auf reine Anwenderprogramme beschränkt hat, zum Teil aber auch an ihrer Kürze und der guten Dokumentation. Außerdem findet der Leser vor allem nützliche Programme. Besonders der Einsteiger wird für diese Art der Hilfestellung dankbar sein.



Das Buch ist in fünf Kapitel unterteilt. Begonnen wird mit Programmierhilfen wie der automatischen Zeilennumerierung und dem Zeilen-Splitting (zerlegt eine Basic-Zeile, in der mehrere Befehle hintereinander stehen, in einzelne Zeilen). Renumber und Delete sind ebenso enthalten wie ein Autostart-File für Diskettenbenutzer. In der Regel handelt es sich dabei um recht kurze Listings, bei denen das Abtippen keinen großen Zeitaufwand erfordert.

Die nächsten drei Kapitel beschäftigen sich mit Textverarbeitung, Grafik und Sound. In diesen Bereichen muß der Leser allerdings mehr Zeit investieren, erhält dafür aber auch Programme, die es in sich haben. Vom kompletten Textverarbeitungsprogramm zum Zeichensatzeditor über vier verschiedene Hardcopy-Routinen wird eine Menge geboten. Dazu gehört auch ein Musikprogramm, das die Atari-Tastatur zur Heimorgel macht. (Über ei-



nen Drucker können sogar Notenblätter ausgedruckt werden.)

Das letzte Kapitel wurde vom Autor als Kramkiste betitelt. Hier findet man extrem kurze Listings, die aber auch nicht ohne Pfiff sind. Das Buch schließt mit einer tabellarischen Gegenüberstellung von ASCII- und internem Code.

Neben dem hohen Nutzwert der Programme bietet die gute Erklärung ihres Aufbaus noch einen nicht zu unterschätzenden Lerneffekt. Somit handelt es sich um ein empfehlenswertes Werk.

Rolf Knorre



ATARI magazin

alte Hefte? – Bitte schön!



___ Ex. 1/87 (6.–)______

Versandkosten (1 Heft 1.40 DM)

Summe______

Alle neuen Leser haben die Möglichkeit, das zurückliegende Heft nachzubestellen. Die Lieferung erfolgt gegen Vorauskasse mit Verrechnungsscheck.

Name

Straße

Ort

Datum

Unterschr.

Bestellschein ausschneiden, ausfüllen, Scheck beilegen und abschicken an:
ATARI magazin
Postfach 1640, 7518 Bretten

●●●● ATARI 800 XL ●●●●
Verk.: Atari-Pascal 150.– DM, Elcomp Forth + Floating-Point + Util. 100.– DM, Atmas II + Toolbox 50.– DM. Anfragen bei Klaus Negle, Hirtenstr. 132, 6757 Waldfischbach.

Atari 800 XL/XE
Suche unterhalts. Cass.-Lernprogr. mit graf. Darstellung für Grund- und Hauptschüler. Ebenso bin ich an anderen Anwendungspr. interessiert. Alfred Bold, Lorenz-Vogel-Weg 6, 7482 Krauchenwies, ☎ 07576/1326

Suche Floppy 1050 für ATARI. Zahle bis 250 DM. ☎ 07802/6290, ab 18 Uhr, nach Bernhard fragen!

Suche dringend Floppy 1050 für Atari, VB 150 DM. ☎ 08152/70339

Verkaufe große XL/XE-Spielesammlung! Billigpreise! Liste (2 DM): M. Enzenberger, Lindenb. 150, 8134 Pöcking

●●● ATARI XL ●●●
Verk.: 800 XL + Floppy 1050 + Bücher + Spiele + Zehnertast. + Datasette + Hefte. ☎ ab 19 Uhr 0911/441154, Preis VS

●●● Drucker ATARI 1029 ●●●
zu verk. 270 DM + Spiel Hotel 20 DM + Draper-Pascal 80 DM + Koronis Rift 30 DM. ☎ 0911/525302

Suche für Atari 800 XL "Steckmodul Basic Computing Language Atari XL 4002" ☎ 09565/2977, verlangt Christian. Bitte erst ab 18 Uhr anrufen. Suche auch Spielmodule für 800 XL

Datenkabel Centr./Centr. und V24/V24 je 30 DM – 10 Brother Typenräder je 35 DM – Taschenrechner Sharp EL-5100 75 DM (NP = 128 DM), 3 Eproms, um Brother IF50 mit Fettdruck aufzurüsten 25 DM. Bernd Kreus, Ringstr. 36, 5100 Aachen, ☎ 0241/520643

ATARI-XL
Verkaufe fast neu Drucker 1029 275.–, Datasette XC11 50.–. Original-Software: ATARI Schreiber 20.–, Time Bandit, Conan, Cavelord, Schreckenstein je 25 DM. F. Stratmann, Duxerstr. 7, 8760 Miltenberg, ☎ 09371/5517, abends

●●● ATARI XL/XE ●●●
Tausche Software auf Disk! Listen an Th. Mayrhans, Vockenthal 2, 8969 Dietmannsried

Suche billige RAM-Erweiterung und Centronics-Anschluß oder Bauanleitung zu 600 XL. Wer kennt den **LASER**-Comp. 310 oder 50? R. Gysin, Albanv. 43, CH-4052 Basel

Orig.-Cassetten abzugeben: Dreibs, Tilt u. Super Cubes, Digger Man, Snokie, Elglide, Slinky, Cap-gold, Steeple Jack, Jet Boat Jack, Darts, Hi-Jack, Dig-Dug, Europas Länder, aufgepaßt je 15 DM. Football Manager u. Mig alle Ace je 25 DM ● Caverns of Khafko = 20 DM ● Bernd Kreus, Ringstr. 36, 5100 Aachen, ☎ 0241/520643

Public Domain Software für ATARI 600, 800, XL + XE Anwender- und Übungsprogramme für Schule, Beruf und Büro. Liste gegen Freiumschlag! Siegfried Sämann, Grenzweg 12, 1000 Berlin 49

ATARI 130 XE
Suche Software auf Diskette wie z.B. Summergames II, Wintergames, Schach, Flugsimulator. Ferner gesucht: Basicerweiterung und Pascal-Compiler. Angebote an M. Schäffer, Hindenburgstr. 234, 4050 Mönchengladbach 1.

●●●● Hey Atari-Freaks!!! **●●●●**
Baue in Eure Floppy Happy ein. Habe Erw. (z.B. 1 Megabyte-Erw.) für Computer und Floppy. Software für Happy. New Games aus USA. Write an H-UM-BAU, Berliner Str. 26, 2250 Husum, ☎ 04841/65533

ATARI 130 XE mit Datasette und viel Zubehör, noch 4 Monate Garantie, für 400.– DM in Originalverp. (NP 600.– DM) ☎ 06151/313778 ab 16.30 Uhr

Suche 1050 bis 200.–! ☎ 07541/22934

ATARI 800 XL: Kaufe, tausche u. verkaufe Spiele, Anwenderprogramme und suche Anleitungen (Kopien). Listen an M. Aalbers, Kappenstr. 113, 4330 Mülheim/Ruhr, ☎ 0208/760690

Suche Cass.-Software für 800 XL. Angebote von Gewerbe u. Privat an: O. Kühne, Siegtalstr. 42, 5227 Herchen

Tausche Programme + Spiele auf Disk für ATARI 800 XL – Udo Brinkmann, Mittellinie 93, 2903 Bad Zwischenahn

Suche für Floppy 1050 passendes Netzgerät. ☎ 0721/408287, ab 18 Uhr

Verkaufe ATARI 800 XL mit Recorder + Software + Joysticks für 200 DM. Sven Vieth, Dahlienstr. 24, 7460 Bal.-Engstlatt, ☎ 07433/20820

Atari 1029 Matrixdrucker mit Hardcopy für 220.–. ☎ 0711/776914

Atari-800XL-Software ab 25 Pf!! Keine Raubkopien. Liste gegen 80 Pf Rückporto bei: Michael Wagner, Altenburgerstr. 64, 6320 Alsfeld. Suche preisw. Matrixdrucker und Floppy.

Verkaufe Kyan-Pascal V2.01, ☎ 06633/1316, ab 18 Uhr

Verkaufe ATARI 800 XL + Recorder + Programme für 300.– DM. 2 Monate alt und wenig gebraucht. ☎ 06432/82881

Public-Domain-Software
Super-PD-Software für Ihren ST zu revolutionären Preisen! Riesenauswahl! Auch auf doppelseitigen Disketten lieferbar! Einfach Gratiskatalog beim PD-Spezialisten anfordern: H. Blankenstein, Ettenhofen 31, 8031 Weßling *G

Bei den mit G bezeichneten Anzeigen handelt es sich um gewerbliche Anbieter.

Suche für ATARI 800 XL: Mule, Mindshadow, PS-Companion, Neverending Story, 221 B Baker-Street, Chessmaster 2000, Sargon 3, Broadsides, Clipper, Bankstreet Writer, Syncalc, DOS 4.0, Business-Manager, Synfile, Ultima 1 + 2, The Pawn!! ●●● Nur schriftl. ●●● K. Schechinger, Robert-Raudner-Str. 9B, 8042 Oberschleißheim

Suche Software für ATARI 800 XL. D. Sauer, Wilhelm-Busch-Str. 28, 4837 Verl 1. Nur Cassette!

Suche Lösungshinweise für die Adventures Mars, Sereamis, Deja-vu, Mask of the Sun. Angebote an: Gabriele Schmitz, Obere Sehlhofstr. 16, 5600 Wuppertal 2

●● ATARI 800 XL ●●
Suche Partner zum Tauschen. Er sollte in Westdeutschland wohnen. Michael Hunz, Fuchsschwanzweg 15, 1000 Berlin 26

!! ATARI: The best forget the rest!!! Ja, nach diesem Motto arbeitet unser Club. Wir haben massig Software und das nur vom Feinsten. Außerdem verscheuern wir günstige Hardware (Happys, Oldrunner, 80-Zeichen). Da ist für jeden etwas dabei. Info?!! Beim ATARI-Club, postlagernd, 8720 Schweinfurt. XL + ST!

Happy-Board = 500% schneller + Backup von geschützten Programmen + Double-Density usw., nur 155.– DM inkl. Porto. ☎ 0212/338537

Suche ATARI-Floppy. ☎ 0203/480682 – Dirk

Floppy 810 Happy 450.–, 1050er Happy Board 160.– DM, Wiesemann Interface 160.– DM. ☎ 04662/5432

●●● Suche für ATARI 800 XL ●●● Amateurfunk-Fernschreib-Programm auf Cassette. Angebot an Michael Klemp, DL7AAZ, Richard-Tauber-Damm 27d, 1000 Berlin 48, ☎ 7423949

Suche Programme für den ATARI 800 XL auf Cassette. Liste an Franz Meier, Sandäcker 2, 8071 Denkendorf

●●●● Atari ●●●●
Verkaufe Atari 130 XE + Disketten + Software für lumpige VHB 899.– (NP: 1099.–) oder tausche 130 XE gegen 800 XL + 200 DM. H. Raber, 7500 Karlsruhe, ☎ 0721/21156, Anruf ab 13 Uhr.

Suche Datei-Textverarbeitungs- und Buchführungsprogramme für 800 XL. Hochwertige Adreßkartei gesucht!!! Zahle Höchstpreise!!! Auf Diskette sehr dringend!!! Werner Windl, Flurstr. 43, 8300 Ergolding, ☎ 0871/76403-74466

∗ ATARI XL/XE ∗ ATARI XL/XE ∗
Speed-Copy-Board für Floppy 1050 (100% Happy-kompatibel) nur DM **149.–**
256K-Erweit. für 800 XL nur DM **169.–**
192K-Erweit. für 130 XE nur DM **89.–**
Weitere Infos oder Bestellungen:
M & P Service, Wilhelm-Böhmert-Str. 4, 2800 Bremen 33 G

■ **DISKETTEN:** ■
■ 5¼", 48 tpi, **DM 0.99,** 2D ■
■ 3½", 135 tpi, **DM 3.19,** 1DD ■
■ 3"-Markendisk. **DM 7.20** ■
■ Auch andere, bes. Garantie! ■
■ Allg. Austro-Agent., Ringstr. 10 ■
■ D-8057 Eching, ☎ 08133/6116 G■

Suche Programme für ATARI ST. Bitte schreibt an: Olaf Spitz, Kappelstr. 2, 5451 Meisbach

---

●●● Software ●●●
Tausche und suche Spiele auf Disk für den Atari 800 XL. Liste an Ralf Bauer, Sauergasse 1, 5420 Lahnstein

---

ATARI 800 XL ●● Suche Grün- oder Bernstein-Monitor für Atari 800 XL. Bitte sofort melden bei: A. Heisinger, Blokkerstr. 29, 5450 Neuwied 22

**● 800 XL ● DATWORK+ ● ATARI ●
Das Dateiprogramm: Masken, blättern, suchen... Turbo-Basic notwendig (wird nicht mitgel.), 48 K, Info 2 DM, Diskette 15 DM. Bernd Thomas, Humpertstr. 17, 5800 Hagen** *G

Suche Software für 800 XL (Disk), suche Happy-Backup Vers. 7.1!! Verk. Atari-Drucker 1025 für 150.– DM. Listen an R. Hirsch, Zitzenbachstr. 60, 5910 Kreuztal 4

ATARI 800! Suche dringend (Diskette): Eine Ultima-Version + Anleitung (Copy). Tausche z.B. gegen Kikstart, Goonies, Mediator, Electra Clide etc. ☎ 06103/43803, Marc verlangen!!!

Atari 800/130
Verkaufe Orig.-Software. Disketten: Vietnam, Leader Board, Screaming Wings, Gemstone-Warrior, mit Originalverpackung + Anleitung!! Alles Total 100.– DM! ☎ 02166/601546 ab 19 Uhr

---

Lichtgriffel nur DM 49,–
komplett mit Programmen + dt. Anleitung
Lieferbar für folgende Computertypen:
Commodore: C 64/C 128/VC 20
Atari: 600XL/800XL/130XE
Schneider: CPC 464/664/6128
Versand gegen Scheck/Nachnahme.
Informationsmaterial gratis!
Bitte Computertyp angeben!
Fa. Klaus Schißlbauer
Postfach 11 71 H, 8458 Sulzbach-Rosenberg
Telefon 09661/6592 bis 21 Uhr

Suche Cass.-Software für Atari 800 XL. Suche außerdem Floppy 1050 für Atari. Zahle bis 200.– DM. Angebote an D. Marchewka, Paul-Löbe-Str. 86, 4390 Gladbeck

---

ATARI XL/XE. Verkaufe Originale: Atlantis 40 DM, Movie Maker 30 DM, Desi. Master 10 DM, alle Disks 70 DM. Wiesemann-72000-Interface 150 DM, Atari Startexter Orig. 40 DM (für XL/XE). Suche Software für ST. Listen an Jürgen Massow, Akazienweg 17, 4905 Spenge, ☎ 05225/9052

Für alle Atari XL/XE
**Deutsches Textadventure mit vielen Räumen. Finden Sie das Zauberschwert, um Ihr Land zu retten. Diskette + 10 DM / 8 Fr an:
Suiss-Soft, M. Schauenberg, Delfterstr. 27, 5004 Aarau (Schweiz)** *G

●●● ATARI 400/800/XL/XE ●●●
Verk. Anwender- & Spielprogramme. Liste gegen 50 Pf Rückporto bei: R. Evertz, Venloer Str. 76, 5024 Pulheim 1 (D/C)

Org. Disks abzugeben: Jumpman, Stealth, Ski-Weltcup, Whist. Brother, Familien-Finanzen, Olympia Lexikon, je 20 DM. Happy Sonderheft = 15 DM, Soloflight 2 = 35 DM ● Soundmachine, Monitor XL; Design-Master je 15 DM, elektr. Schaltungen konstr. = 25 DM ●
Bernd Kreus, Ringstr. 36, 5100 Aachen, ☎ 0241/520643

● XL/XE ● !! ATARI !! ● XL/XE ●
Suche Summergames II + Wintergames + Software jeglicher Art! Schickt Eure Listen an Markus Vogel, Kronenstr. 40, 3000 Hannover 1 oder ruft unter ☎ 0511/332349 an!!

(800 XL) ●●● ATARI ●●● (800 XL)
Die Super-Spieledisk ist da!! 14 Spiele! Schickt 20-DM-Schein an Wolfgang Kröger, Am Buchenbaum 14, 3540 Korbach

---

Für alle ST: Public-Domain-Software, Freiprogramme auf 60 Disks, 400.– DM; einzeln 8.– DM. Freiliste:
PDS E. Behrendt, Kudowastr. 23A, 1000 Berlin 33 *G

Hübsche Mädchen, Raumschiffe, Könige und vieles mehr gibt es schon für Ihren Atari ST auf PD-Software zu Spottpreisen!!! Riesige Auswahl und schnelle Lieferzeit! Gratisinfo bei Ralf Markert, Balbachtalstr. 71, 6970 Lauda-Oberbalbach. Es lohnt sich!! *G

Public Domain für ST
Jede 1S-Disk ab 6.00 DM (inkl. Disk),
Jede 2S-Disk ab 8.50 DM (inkl. Disk).
Info gegen Freiumschlag bei:
Krieger & Fette GbR, Bahnstraße 23a, 5600 Wuppertal 11, ☎ 0202/789945 *G

●●● ATARI ST ●●●
Spiele, Musik, Grafik, Anwendungen. Gratisinfo bei Edward Twardoch, Kais.-Wilh.-Str. 88, 1000 Berlin 46 *G

---

40 MByte PD-Software für den ST haben wir schon, und es wird immer mehr! 1000 Programme aller Art fast zum Nulltarif! Gratiskatalog anfordern bei G. Köhler, Mühlgasse 6, 6991 Igersheim, ☎ 07931/44661 *G

●●● ATARI ST ●●●
Verk. günstig nagelneue ST-Floppy SF 354!! Preis VS. ☎ ab 19.00 Uhr: 0911/441154
●●● ATARI ST ●●●

●● F.E.T. Farb-Videodigitizer ●●
Digitalisiert das Farbsignal einer Videokamera in 16 Echtfarben. 528.– DM für Atari 520/1040 ST. Creative Video, Postfach 1501, 8520 Erlangen, ☎ 09195/2728 *G

PD-Software für Ihren Atari ST!
Über 80 Disketten voll mit Spielen, Zeichen- und Grafikprogrammen und vieles mehr zu sagenhaft günstigen Preisen!! Gratisinfo bei Ralf Markert, Balbachtalstr. 71, 6970 Lauda *G

ATARI 260 ST: 1 Megabyte Aufrüstung, steckbare Platine ohne Lötarbeiten, Epromsteckkarte und User-Port-Karte. **Für 400/800:** OS Rev. B und es laufen alle Programme. **Happy 1050** und High-chip. ☎ 0202/402814

Suche 520 STM-Kontakte oder andere ST-Besitzer und Tauschpartner für ST-Programme. Markus Theimer, Schwedenstr. 2, 6203 Hochheim

---

● **ST ● SUCHE ●ST** ●
Kontakt zu ST-Usern zwecks Software- und Erfahrungsaustausch! Zahle gut! Schickt Eure Listen an: Stephan Daniaux, Murstraße 77, A-6064 Rum/Tirol, ☎ 64184

Macintosh-Emulator für Atari-ST. Ermöglicht, Mac-Software "zu fahren". Dito ebenfalls mit Uhr (ST + Mac), sehr kompatibel + 20% schneller!! Bussystem, RAM-Erweiterungen, ROMs, 1-MB-Drives. ☎ 02871/183489

ST Suche Software! ST
ST Suche Tauschpartner! ST
ST Suche Harddisk bis 500.– DM! ST
ST Suche Speicher für 260 ST! ST
ST Suche Bücher über/von ST ST
ST Lutz Martschin, Reherweg 5a, ST
ST 3258 Aerzen 1 ST

●● ATARI XL/XE-Software ●●
Verkaufe für 20 DM folgende Programmarten auf Diskette (etwa 35 Stück):
1. Lernprogramme (Sprachen E/F/S/L),
2. Lernprogramme (Mathe/Physik/Ch.)
3. Software (Spiele)!! Schein an: J. Schwarzer, Th.-Heuss-37, 8660 Münchberg. Die ersten 25 Besteller: 1 Geschenk

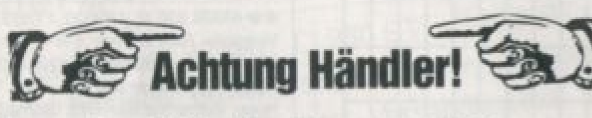
---

ATARI 600/800/130 XL/XE: Verkaufe infolge Umstellung auf ST-Serie alle meine kommerziellen 8-Bit-Programme zu Superpreisen! Bis 85% unter NP! Liste anfordern bei: Marco J. Meyer, Kreuzmattweg 15, CH-4144 Arlesheim

---

Atari 800 XL: Wer sendet mir zu "The Goonies" die Spielanleitung? Andreas Krebs, Bahnhofsiedlung 1, 8941 Buxheim

---



●●● ATARI 800 XL (Disk) ●●●
Suche Tauschpartner, habe u.a.: Ninja, Ballblazer, River Raid, Pen-90, Archon, Gremlins. ☎ 08851/5553

Verkaufe Spiele- + Grafik-Programme für 800 XL + 130 XE. Listings ab 0.50 DM, Spiele ab 1.50 DM plus 5 DM für Diskette. Suche Est. '86 + Video-Archiv. Liste gegen 1.20 DM Rückporto bei Helmut Nagel, Josef-Maria-Olbrich-Str. 34, 4000 Düsseldorf 13

ATARI XL/XE ●● Suche: DSS-Module!! Suche Floppy 1050 (180 DM!!). M. Schubbert, Musfeldstr. 77, 4100 Duisburg 1

Suche das Programm Graph It auf Diskette. DXG 5717. Zahle gut. Jürgen Rost, Hainbuchenweg 6, 7600 Offenburg 27. ☎ 0781/24746 ab 19 Uhr

● Lohn- und Einkommensteuer ● Super Jahresausgleich + Steuererkl.-wahl + Rentnerertrag + Analyse! Atari, Sinclair, Commodore, TI 99/4A. Jährl. Aktualisierung 10.– DM. Cass. 60.–, Disk. 70.– DM. Info: Horst Ilchen, Niederfelder Str. 44, 8072 Manching, ☎ 08459/1669 G

Verkaufe: Riteman F+ mit Interface für Atari (Centronics), 100% Epson-komp., 9×9 Matrix, NP 1100.– DM, ½ Jahr alt, VP: 600.– DM. ☎ 02625/4612

Verk. 800 XL, 128-K-RAM + Floppy 1050 + zwei Joysticks + Software (Bruce Lee, Chess, Summergames, Int. Karate, Cyrus u.a.), zusammen VB 600 DM. Bei ☎ 05421/1877. Verk. außerdem Soundmachine, Hexenküche je 15 DM und eine Tischorgel PS-3 von Yamaha: 475 DM (9 Instr. u. 4 Rhythmen)

● 1050 ● Floppy-Speeder ● 1050 ● Happy 1050 zu verkaufen. Bis 500% schneller. Backup von jeder Disk. Für 145.– DM plus Porto abzugeben. ☎ 02377/1076 (4269)

800 XL – Suche Tauschpartner (nur Disk). Liste an T. Hügler, Prager Str. 14, 5060 Berg. Gladbach 1. Suche gutes Kopierprogramm + Akustikkoppler bis 200 DM

Suche gute Kopierprogramme (Disk) und Hitchhikers Guide to the Galaxy für Atari. Angebote an H. Schmitt, Sonnenstr. 76, 4600 Dortmund 1

●●●● Atari-800-XL-User ●●●● Ich suche für meinen Atari 800 XL Module, Bücher, Listings, Programme auf Cassette und einen günstigen Drucker. Bitte sendet Eure Angebote an F. Kugler, Kamblickweg 39, 4300 Essen 13

Tausche / verkaufe Software (D + C) W. Weiershausen, Hesseneckstr. 12, 6340 Dillenburg 1. ☎ 02771/21366

●●● ATARI ●●●
Atari 800 XL + Atari 1010 beide mit Staubschutzhaube + Software auf Cassette + Leercassetten. Telefon angeben. Preis VS. Dieter Kick, Weberweg 2, 8590 Marktredwitz

ATARI ●●● RABE I ●●● ATARI
Ein Programmier-, Testunterstützungs-, Fehlersuch- und Programmanalysiersystem ermöglicht Einblicke in listgeschützte Software. Per Tastendruck jederzeit aufrufbar, Pr. 79.– DM, EPROM + Disk. Info bei A. Kern, Pilartzstr. 5, 8091 Bachmehring, ☎ 08071/1225

● ATARI ● ATARI ● ATARI ● ATARI ●
● Die neue Superdiskette mit 10 Superspielen ist da. Auch 50 Spiele zu 30 DM oder 100 Spiele zu 50 DM. Nur auf Disk.
● Verkaufe auch Bootdisks zu je 10 DM. Z.B. Strip-Poker, The Quest, Kaiser usw. ☎ 0541/37787. Ruft an!

Suche Cass.-Software für ATARI 800 XL. Angebote an Sven Burmeister, Lessingstr. 27, Helmstedt

● XL ●● SPACE-BUMPERS ●● XE ●
Das Super-Actionspiel für den XL/XE, 100% Maschine!!! Auf Disk nur 20 DM + Bonus-Soft!! Info gratis oder 20 DM an: Damata Software, Markus Kretzer, v. Stauffenberg-Str. 32, 2120 Lüneburg G

●● ATARI 600 XL / 800 XL / 130 XE ●●
Verkaufe Teile meiner SW-Sammlung! Movie Movie, Explorer 2 je 10.– DM. Die Flucht, Rote Diamant 10.– DM. Super Huey, Vietnam je 50.– DM. Ultima 3, Ultima 4 je 80.– DM. Lutz Martschin, Reherweg 5, 3258 Aerzen 1, ☎ 05154/1495

Suche Spiele für ATARI 600 XL. Angebote an Olaf Albert, Ibbingsen 1A, 5758 Fröndenberg-Frömern, ☎ 02378/4987

Bei den mit G bezeichneten Anzeigen handelt es sich um gewerbliche Anbieter.

Bestellschein für Kleinanzeigen

Bitte veröffentlichen Sie in der nächsten Ausgabe folgende Anzeige:

Meine Anzeige soll in 1 ☐ 2 ☐ 3 ☐ Ausgaben erscheinen.

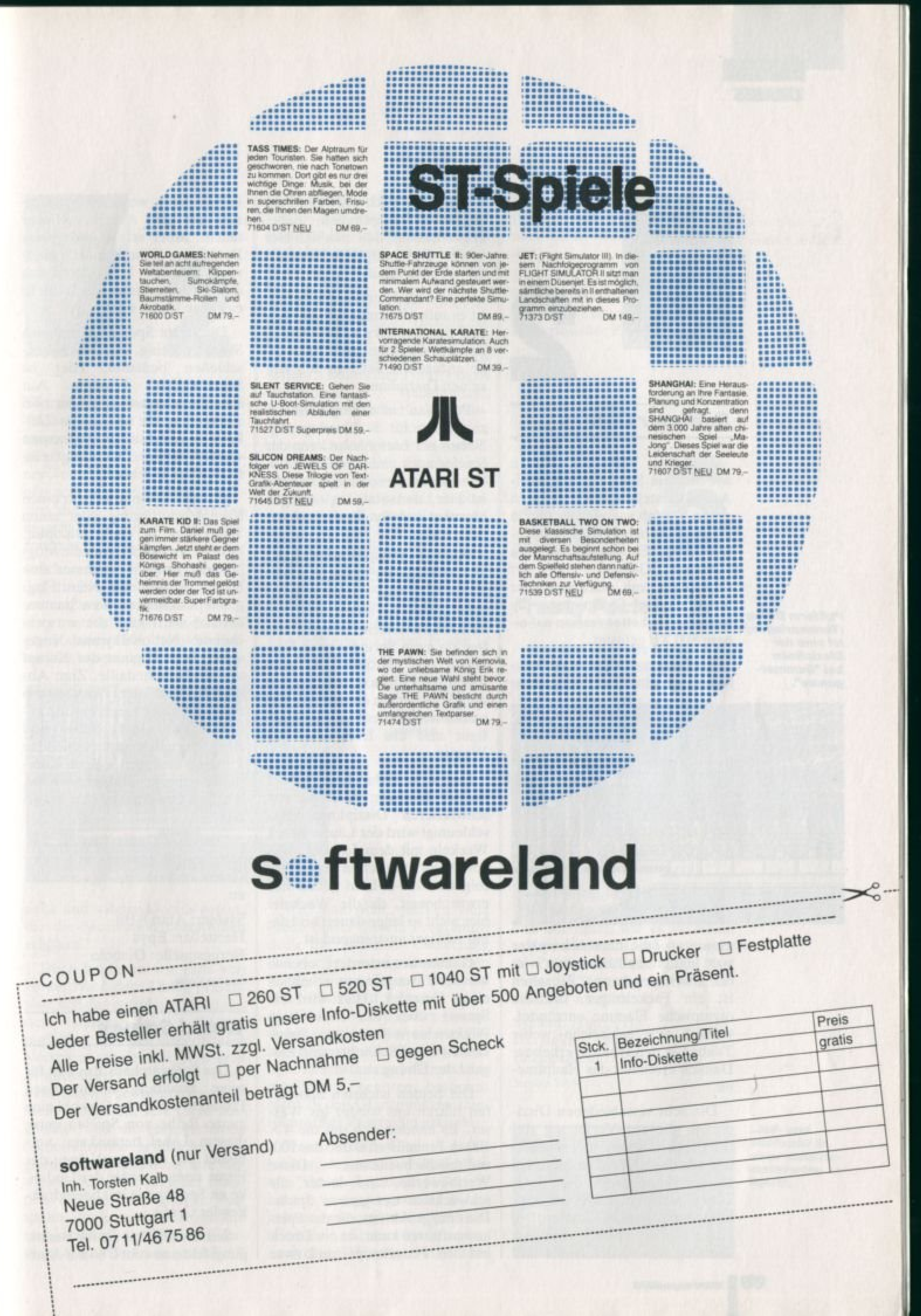
Vor- und Zuname

Straße

PLZ/Ort

Datum Unterschrift

Bitte ankreuzen:
☐ **Private Kleinanzeige: pro Zeile 1,– DM**
☐ **Gewerbliche Kleinanzeige: pro mm 2.40 DM + 14% MwSt**
Bei mehreren Kleinanzeigen bitte Bestellschein kopieren. Den Betrag in Briefmarken oder als Scheck zusammen mit der Kleinanzeige einsenden. Bei gewerblichen Kleinanzeigen ist keine Vorauszahlung notwendig. Hier erhalten Sie nach Abdruck eine Rechnung.
ATARI *magazin*
Redaktion
Postfach 1640, 7518 Bretten

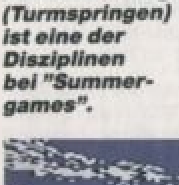

ST-Spiele
TASS TIMES: Der Alptraum für jeden Touristen. Sie hatten sich geschworen, nie nach Tonetown zu kommen. Dort gibt es nur drei wichtige Dinge: Musik, bei der Ihnen die Ohren abfliegen, Mode in superschrillen Farben, Frisuren, die Ihnen den Magen umdrehen.
71604 D/ST NEU DM 69,–
WORLD GAMES: Nehmen Sie teil an acht aufregenden Weltabenteuern: Klippentauchen, Sumokämpfe, Stierreiten, Ski-Slalom, Baumstämme-Rollen und Akrobatik.
71600 D/ST DM 79,–
SPACE SHUTTLE II: 90er-Jahre. Shuttle-Fahrzeuge können von jedem Punkt der Erde starten und mit minimalem Aufwand gesteuert werden. Wer wird der nächste Shuttle-Commandant? Eine perfekte Simulation.
71675 D/ST DM 89,–
JET: (Flight Simulator III). In diesem Nachfolgeprogramm von FLIGHT SIMULATOR II sitzt man in einem Düsenjet. Es ist möglich, sämtliche bereits in II enthaltenen Landschaften mit in dieses Programm einzubeziehen.
71373 D/ST DM 149,–
INTERNATIONAL KARATE: Hervorragende Karatesimulation. Auch für 2 Spieler. Wettkämpfe an 8 verschiedenen Schauplätzen.
71490 D/ST DM 39,–
SILENT SERVICE: Gehen Sie auf Tauchstation. Eine fantastische U-Boot-Simulation mit den realistischen Abläufen einer Tauchfahrt.
71527 D/ST Superpreis DM 59,–
SILICON DREAMS: Der Nachfolger von JEWELS OF DARKNESS. Diese Trilogie von Text-Grafik-Abenteuer spielt in der Welt der Zukunft.
71645 D/ST NEU DM 59,–
ATARI ST
SHANGHAI: Eine Herausforderung an Ihre Fantasie. Planung und Konzentration sind gefragt, denn SHANGHAI basiert auf dem 3.000 Jahre alten chinesischen Spiel „Ma-Jong". Dieses Spiel war die Leidenschaft der Seeleute und Krieger.
71607 D/ST NEU DM 79,–
KARATE KID II: Das Spiel zum Film. Daniel muß gegen immer stärkere Gegner kämpfen. Jetzt steht er dem Bösewicht im Palast des Königs Shohashi gegenüber. Hier muß das Geheimnis der Trommel gelöst werden oder der Tod ist unvermeidbar. Super Farbgrafik.
71676 D/ST DM 79,–
BASKETBALL TWO ON TWO: Diese klassische Simulation ist mit diversen Besonderheiten ausgelegt. Es beginnt schon bei der Mannschaftsaufstellung. Auf dem Spielfeld stehen dann natürlich alle Offensiv- und Defensiv-Techniken zur Verfügung.
71539 D/ST NEU DM 69,–
THE PAWN: Sie befinden sich in der mystischen Welt von Kernovia, wo der unliebsame König Erik regiert. Eine neue Wahl steht bevor. Die unterhaltsame und amüsante Sage THE PAWN besticht durch außerordentliche Grafik und einen umfangreichen Textparser.
71474 D/ST DM 79,–
softwareland
COUPON
Ich habe einen ATARI □ 260 ST □ 520 ST □ 1040 ST mit □ Joystick □ Drucker □ Festplatte
Jeder Besteller erhält gratis unsere Info-Diskette mit über 500 Angeboten und ein Präsent.
Alle Preise inkl. MWSt. zzgl. Versandkosten
Der Versand erfolgt □ per Nachnahme □ gegen Scheck
Der Versandkostenanteil beträgt DM 5,–
Stck. | Bezeichnung/Titel | Preis
1 | Info-Diskette | gratis
Absender:
softwareland (nur Versand)
Inh. Torsten Kalb
Neue Straße 48
7000 Stuttgart 1
Tel. 07 11/46 75 86



2

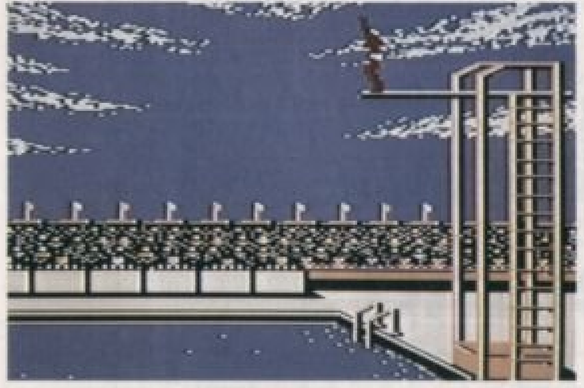
Summer Games

Mit etwas Verspätung wollen wir hier ein Spiel für alle 8-Bit-Ataris vorstellen, das sich durch große Vielfalt auszeichnet und in keiner Programmbibliothek fehlen sollte. Es handelt sich um "Summer Games", eine Art Olympiade mit teilweise eher ausgefallenen Disziplinen. Ich habe die Diskettenversion auf einem 800 XL getestet.



Platform Diving (Turmspringen) ist eine der Disziplinen bei "Summergames".

Zu Beginn des Spiels läuft automatisch eine Eröffnungszeremonie ab (die man sich später vom Hauptmenü aus immer wieder anschauen kann). Zu sehen ist ein Fackelträger, der die olympische Flamme entzündet, worauf sich zahlreiche weiße Tauben in den Himmel erheben. Danach erscheint das Hauptmenü.



Die acht verschiedenen Disziplinen, die zur Verfügung stehen, lassen sich auf mehrere Arten angehen. Möglich sind ein Komplettdurchgang, der Wettkampf in nur einer Sportart oder Übungen in einer Disziplin. Die letzte Option erlaubt eine Art Training ohne Konkurrenz und Wertung, was ich als besonders angenehm empfunden habe. Hinzu kommt, daß man sich hier in aller Ruhe sämtliche Sportarten ansehen kann.

Aus dem Hauptmenü heraus ist es außerdem möglich, die Anzahl der Joysticks zu bestimmen und sich die aktuellen Weltrekorde anzusehen. Jetzt zu den einzelnen Disziplinen:

Pole Vault ist die englische Bezeichnung für Stabhochsprung. Neben der Lattenhöhe kann der Spieler auch einen von drei Stäben wählen. Besonders schwierig ist hier, den Stab im richtigen Moment auf den Boden zu setzen.

Platform Diving nennt sich die zweite Disziplin. Dahinter versteckt sich das Turmspringen. Hier konnte ich bisher noch keine nennenswerten Erfolge erzielen, da mein Springer nach Verlassen der Plattform nur wild durch die Luft flog, was ihm leider keine Punkte einbrachte. Bewertet werden der Flug der Spielfigur und das Eintauchen ins Wasser.

Der 4 × 400-m-Lauf gehört meiner Meinung nach zu den schwächeren Disziplinen. Beschleunigt wird der Läufer durch Wackeln mit dem Joystick, was auf Dauer ganz schön nervt. Der folgende 100-m-Lauf gefällt mir etwas besser, da die Wackelei hier nicht so lange dauert und damit weniger anstrengend ist.

Gymnastics erfordert, ebenso wie das Turmspringen, erheblich mehr Geschick. Hier wird ein Sprung über das Pferd von Punktrichtern bewertet. Dabei kommt es auf Schönheit und Eleganz der Übung an.

Die beiden nächsten Sportarten führen uns wieder ins Wasser. Es handelt sich um die 4 × 100-m-Freistilstaffel und das 100-m-Freistilschwimmen. Diese Wettbewerbe sind leider die schwächsten des ganzen Spiels. Die einzige Aktion, die der Spieler ausführen kann, ist ein Druck auf den Feuerknopf, und zwar immer dann, wenn der Schwimmer mit dem Arm ins Wasser taucht. Aber selbst ohne diese Betätigung kommt man irgendwann ans Ziel. Hier hätten sich die Programmierer doch mehr einfallen lassen sollen.

Die letzte Sportart nennt sich Skeet Shooting, was Tontaubenschießen bedeutet. Hier ist Schnelligkeit Trumpf. Auf Knopfdruck starten ein oder zwei Tontauben, die über einen Zielkreis anvisiert und abgeschossen werden sollen. Das klingt einfach, ist es aber nicht.

Entscheidet man sich für einen Komplettdurchgang oder einen Wettkampf in einer Disziplin, besteht vor Spielbeginn die Möglichkeit, jedem Teilnehmer eine Nation zuzuordnen. Neun Flaggen repräsentieren diese Staaten. Danach wird kurz die entsprechende Nationalhymne angespielt. Nun beginnt der Kampf um die Goldmedaille. Zum Abschluß wird der Wettbewerb noch mit einer Siegerehrung gekrönt. Für Gold, Silber und Bronze erhält man verschiedene Punkte, die nach einem Komplettdurchgang addiert werden, um den Gesamtsieger zu ermitteln.

"Summer Games" ist ein aufwendiges Programm mit guter Grafik und vielen Gags und Ideen.

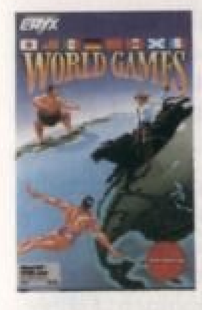
System: Atari 8 Bit
Hersteller: Epyx
Bezugsquelle: Diabolo

Stephan König

World Games

Das Softwarehaus Epyx ist für seine qualitativ hochwertigen Produkte berühmt. Obwohl eine ganze Reihe von Spielen unter diesem Label herauskam, verbindet sich mit dem Namen der Firma immer wieder der Gedanke an Sportspiele mit beeindruckender Grafik.

Seit kurzem hat sie ihr Betätigungsfeld nun vom C 64 auf Atari



2

ST und Amiga ausgedehnt. Nach "Winter Games" liegt jetzt das neueste Produkt der Sportspielreihe gleich für den 16-Bit-Rechner vor. "World Games" vereinigt acht sehr originelle Disziplinen, die als Traditionssportarten in acht Ländern der Erde bezeichnet werden.

Schon hier ist zu erkennen, daß das Programm zwar als Fortsetzung bezeichnet werden kann, aber trotzdem aus dem bisherigen Rahmen fällt, denn die Entwickler haben viel Wert auf originelle und witzige Details gelegt, die die Motivation beträchtlich erhöhen.

Der Gewichtheber steht bereit. Lassen Sie ihn nicht warten.

Doch beginnen wir ganz von vorne. Nach dem Laden präsentiert sich das gewohnte Menü aus der "Games"-Reihe; diesmal muß man auf einen farbenfrohen Vorspann verzichten. Nun geht es mitten ins Geschehen. Nachdem man wie gewohnt seinen Namen angegeben, sein Land ausgewählt und die entsprechende Nationalhymne gehört hat, beginnt der Wettbewerb mit Gewichtheben. Wie bei jeder Disziplin erscheint ein kleiner Vorspann, der kurz über Geschichte und Herkunft der Sportart informiert. Dann darf man seine Kraft unter Beweis stellen. Um hier erfolgreich zu sein, gilt es nicht so sehr, am Joystick zu rütteln, vielmehr braucht man eine ausgefeilte Technik im Stemmen und Heben der Hantel.

Auch die restlichen Disziplinen verlangen mehr Technik als einen stabilen Joystick. Faßspringen, Klippentauchen, Skislalom, Balancieren auf einem Baumstamm, Bullenreiten, Baumstammwerfen und Sumo-Ringen sind die Sportarten, in denen sich der Spieler üben muß.

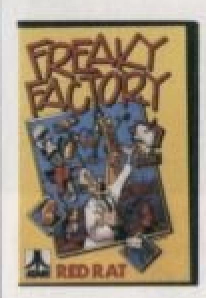
In vielen Fällen ist aber ein Fehlschlag fast interessanter als der Erfolg. Der auf und ab hüpfende Schotte, dem ein Baumstamm auf den Fuß gefallen ist, das rote Gesicht des Gewichthebers, der die Hantel zu lange halten muß oder der verdutzte Blick des Faßspringers, der etwas früher landet als geplant – das alles ist schon einen kleinen Fehler in der Steuerung wert.

Wie man sieht, stecken viele Ideen hinter diesem Programm, doch die grafische Umsetzung läßt noch ein wenig zu wünschen übrig. Einige Disziplinen sind zwar relativ gut gelungen (z.B. das Gewichtheben), doch bei anderen beginnt man, an den Fähigkeiten der Grafiker zu zweifeln. Baumstammwerfen und Sumo-Ringen gehören bestimmt nicht zu den Glanzpunkten amerikanischer Programmierkunst.

Wem "Winter Games" gefallen hat, der wird an "World Games" dennoch seine reine Freude haben. Ein witziges und unterhaltsames Sportspiel erwirbt man allemal. Außerdem bleibt immer noch eine Hoffnung: Das nächste Spiel dieser Reihe kommt bestimmt, und vielleicht benutzt man dann bessere Grafikroutinen.

System: Atari ST 512 KByte, Farbmonitor
Hersteller: Epyx
Bezugsquelle: Profisoft

Thomas Tai





Freaky Factory

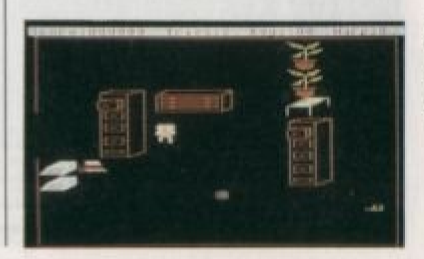
Red Rat Software hat ein neues Programm herausgebracht. Es nennt sich "Freaky Factory" und läuft auf allen 8-Bit-Ataris.

In einer vollautomatisierten Fabrik der Zukunft haben schon längst Roboter alle menschlichen Arbeitskräfte verdrängt. Als der Zentralcomputer durchdreht, laufen alle Roboter Amok. Der Spieler schlüpft in die Rolle des Wissenschaftlers, der alles wieder in Ordnung bringen soll.

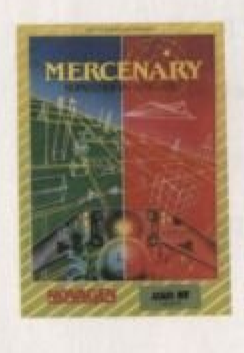
Im Prinzip handelt es sich bei "Freaky Factory" um ein Lauf- und Suchspiel, wie es sie schon massenweise gibt. Auch die große Zahl der Räume (über 200) kann das Programm nicht interessanter machen.

System: Atari 8 Bit
Hersteller: Red Rat
Bezugsquelle: Compy-Shop

Stephan König



Lauf- und Suchspiel in einer vollautomatisierten Fabrik.



2

Mercenary

Dieses schon fast klassisch zu nennende Science-fiction-Spiel gibt es jetzt auch für die ST-Computer. Voraussetzung ist allerdings ein Farbmonitor bzw. ein entsprechender Modulator.

Auf dem Cover findet man unter dem Spieltitel den Vermerk "Kompendium Ausgabe". Dahinter verbirgt sich die Tatsache, daß auf der Diskette eigentlich zwei Spiele abgespeichert sind, "Flucht von Targ" und "Die zweite Stadt". Letzteres kann als Fortsetzung des ersten betrachtet werden. Von Aufbau und Grafik her sind beide weitgehend identisch. (Bei "Die zweite Stadt" wurden andere Farben eingesetzt, da die Handlung in der Dämmerung des roten Winters spielt.) Gedacht ist die Fortsetzung hauptsächlich für Spieler die Teil 1 bereits in- und auswendig kennen. Hier treffen sie auf neue Gefahren und müssen andere Aufgaben lösen.

Worum geht es nun bei "Mercenary"? Die Spielhandlung findet auf und unter der Oberfläche des Planeten Targ statt. Dort regieren die heimtückischen Mechanoiden, die die ursprünglichen Bewohner, Palyaren genannt, aus ihrer Heimat vertrieben haben. Diese umkreisen nun den Planeten mit einem Kolonieschiff und sinnen auf Rache.

Der Spieler hat mit diesen Machtkämpfen eigentlich nichts zu tun, da er mit seinem interstellaren Raumschiff Prestinium nur die Gegend erkunden will. Leider läuft einiges schief, und die Prestinium macht eine Bruchlandung auf Targ. Dabei wird sie völlig zerstört, und der Spieler befindet sich mitten im Geschehen. Zur Unterstützung ist ihm nur der Bordroboter Benson geblieben, der ihm hin und wieder Informationen liefert.

Ziel des Spiels ist die Flucht von Targ. Sie ist natürlich nur möglich, wenn sich ein vernünftiges Raumschiff beschaffen läßt. Dieses kann man kaufen oder stehlen. So schnell geht das aber nicht. Auf der Planetenoberfläche sind zahlreiche Objekte und Gegenstände zu sehen. Glücklicherweise kann man gleich zu Anfang einen Gleiter kaufen, der die Fortbewegung erheblich erleichtert. (Ohne das Gefährt muß man sich zu Fuß auf den Weg machen, was wenig empfehlenswert ist.)

Alleine das Überfliegen von Targ ist die Anschaffung des Programms wert. Tolle 3-D-Vektorgrafiken und andere Effekte sorgen für gute Unterhaltung. "Mercenary" ist kein hektisches Spiel, in dem in Sekundenbruchteilen alles entschieden wird. Hier kann man tatsächlich unbegrenzt lange herumfliegen, ohne daß etwas Besonderes passiert. In der Anleitung wird darauf hingewiesen, daß es kein eigentliches Ende gibt, der Spieler verliert also nicht sein Leben. Schlimmstenfalls wird der Gleiter abgeschossen.

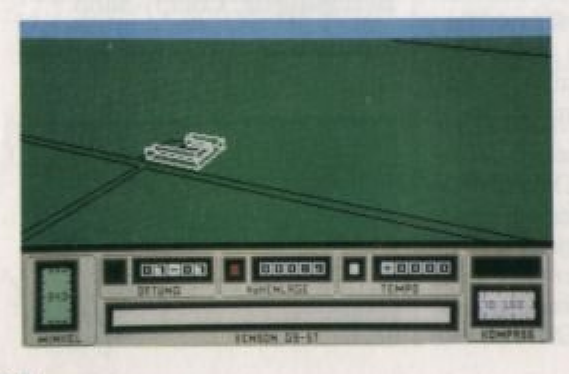
An verschiedenen Punkten der Planetenoberfläche befinden sich Aufzüge, die in die Unterwelt führen. Dort leben und arbeiten die Mechanoiden. Gänge, Räume usw. sind auch hier in dreidimensionaler Grafik dargestellt.

Jetzt kann man Kontakt aufnehmen, in die Dienste der Mechanoiden treten, mit ihnen handeln und vieles mehr. Es geht ja darum, möglichst viel Geld aufzutreiben, um ein neues Raumschiff zu kaufen. Hat man z.B. den Energiekristall aufgenommen, der in einem der Räume der ersten Etage liegt, kann man diesen gegen 100.000 Punkte eintauschen. Bei Spielbeginn hat man lediglich 9000 Punkte. Wer will, kann sich auch auf Kampfhandlungen einlassen.

"Mercenary" ist viel zu komplex, um es in wenigen Worten umfassend zu beschreiben. Es ist ein Spiel für lange Abende und unangenehme Regentage. Mit ein wenig Fantasie kann man sich richtig in die Rolle des gestrandeten Abenteurers hineinversetzen. Da Anleitung und Bildschirmkommentare ins Deutsche übersetzt wurden, treten auch keine Kommunikationsschwierigkeiten auf.

Der Diskette liegt übrigens noch ein verschlossener Umschlag bei, der als Überlebenspackung bezeichnet wird. Hier findet man einige Karten und Erklärungen, die das Spiel erleichtern. Das gilt aber nur für "Flucht von Targ". "Die zweite Stadt" muß im Alleingang bewältigt werden.

Ich halte "Mercenary" für ein hervorragendes Programm, nicht nur im Hinblick auf die Gestaltung, sondern auch was die Spielmotivation und die Idee betrifft. Getestet wurde das Spiel auf einem ST; es gibt aber auch eine Version für die kleinen Ataris.

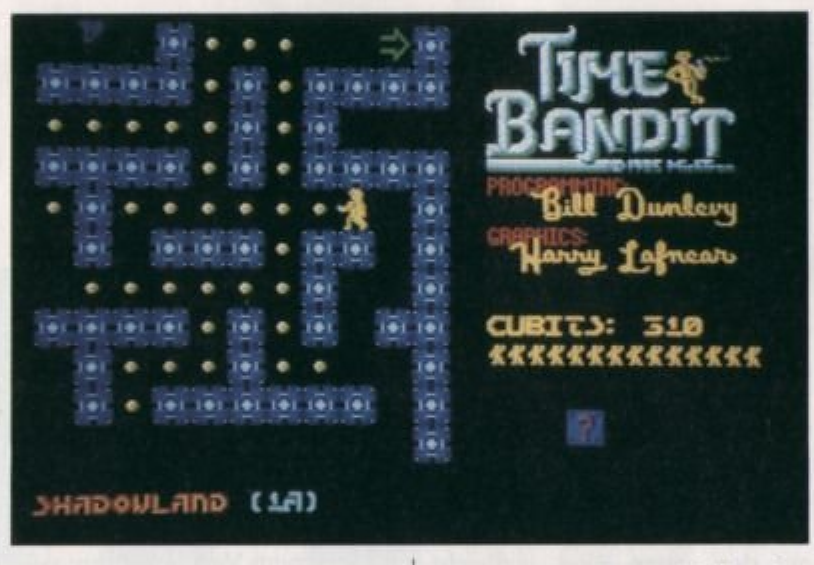


Raum-Schiffbruch auf dem Planeten Targ ist der Ausgangspunkt von "Mercenary".

System: Atari ST,
Farbmonitor
Hersteller: Novagen
Bezugsquelle: Diabolo

Rolf Knorre

Time Bandit

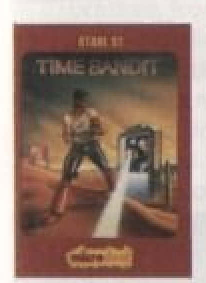


Verwirrung in Zeit und Raum. In "Time Bandit" schlägt man sich durch verschiedene Zeitzonen.

Dieses Programm stellt meiner Meinung nach eine optimale Mischung zwischen Action-Spiel und Adventure dar, wobei die Action-Elemente allerdings überwiegen. Dabei ist es weniger eine tolle Story als vielmehr die Umsetzung, die mich begeistert. Im Grunde handelt es sich um ein typisches High-Score-Spiel, in dessen Verlauf man Jagd auf Punkte macht.

Gestartet wird immer in einer parkähnlichen Landschaft, in der noch nicht viel passiert. Hier sieht man 16 verschiedene Symbole, die jeweils eine eigene Umgebung und Zeitzone darstellen. Jede dieser Zonen hat wiederum 16 verschiedene Levels. Man kann sich jetzt ein Symbol aussuchen, die Spielfigur darauf führen, und schon wechselt das Bild.

Die 16 Zeitzonen im einzelnen zu erläutern, ist fast unmöglich. Da gibt es z.B. futuristisch gestaltete Technikbereiche, altertümliche Burgen, ein komplettes "PacMan"-Spiel und vieles mehr. Die Grafik ist sehr detailliert und liebevoll gestaltet.



2

Jede Zone besitzt nur einen Ausgang, der jedoch verschlossen ist. Irgendwo liegt nun der passende Schlüssel. Es gilt also, diesen zu finden, auf dem Weg Punkte zu machen und möglichst unbeschadet in den Park zurückzukehren.

Glücklicherweise hat man gleich zu Anfang 10 Leben zur Verfügung, die auch dringend erforderlich sind. In den Zonen wimmelt es von allen möglichen Gestalten. Schon eine kurze Berührung kostet ein Leben. Hat man 1000 Punkte erreicht, erhält man aber ein Zusatzleben. Die Spielfigur ist manchmal bewaffnet, manchmal auch nicht. Dazu kommt, daß alles sehr schnell abläuft.

In verschiedenen Zeitzonen tauchen Symbole auf, die dem Spieler unter Umständen sehr hilfreich sein können. Berührt man ein solches Symbol, wird die Grafik gelöscht. Auf dem Bildschirm erscheint dann ein Text, wie man ihn von herkömmlichen Adventures kennt. Je nach Lage der Dinge lassen sich nun besondere Aktionen einleiten oder nur Informationen erhalten. Darauf kann man übrigens auch verzichten, indem man den Ort wieder verläßt. Das Spiel geht dann an der Stelle weiter, an der man es unterbrochen hatte.

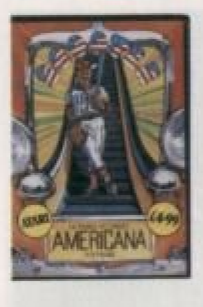
Neben den feindlichen Gestalten und den Hinweissymbolen gibt es auch noch Löcher und Leitern, Einwegstraßen usw. Das Ganze ist wie ein Labyrinth aufgebaut, von dem immer nur ein kleiner Teil zu sehen ist. Hier könnte man Kritik üben, da die Grafik nur rund die Hälfte der Bildfläche des Monitors einnimmt. Der Rest ist mit dem Programmtitel, den verbleibenden Leben und einigen Zusatzinformationen belegt. Diesen Teil hätte man ruhig kleiner halten können.

"Time Bandit" lebt von den Überraschungen, denen man in den verschiedenen Zonen und Levels begegnet. Das Programm bietet als Besonderheit noch eine Two-Player-Option an. So können zwei Personen über Joysticks oder die Tastatur miteinander spielen (und nicht gegeneinander, wie es sonst üblich ist). Zu diesem Zweck wird die Grafik in zwei Fenster aufgeteilt. Jetzt laufen zwei Gestalten durch die Landschaft, die völlig unabhängig voneinander agieren können. Kommt einer der beiden um, kann er als Schatten immer noch die Gegend erkunden und seinem Mitspieler helfen. Diese Möglichkeit finde ich besonders gelungen.

Zum Abschluß ein Spieltip: Als Anfänger sollte man vielleicht mit "PacMan" beginnen, da man hier die meisten Punkte und Zusatzleben sammeln kann, ohne sich besonders in Gefahr zu begeben. Die Kennzeichnung dieser Zone im Park ist völlig eindeutig.

System: Atari ST, Farbmonitor
Hersteller: Microdeal
Bezugsquelle: Diabolo

Stephan König

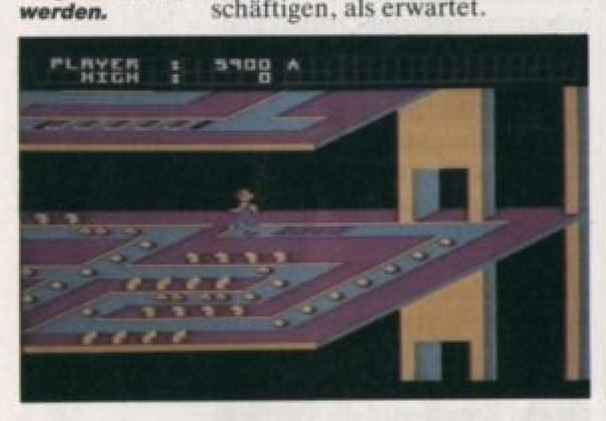


3

Scooter

Wer knifflige Actionspiele mag, wird von "Scooter" begeistert sein. Auch wenn das Spiel auf den ersten Blick vielleicht einfach erscheint, kann man sich doch erheblich länger damit beschäftigen, als erwartet.

Pillen über acht Stockwerke. Wie bei Pacman müssen sie aufgesammelt werden.



Mit dem Coverbild hat das Programm allerdings überhaupt nichts zu tun. Der dort abgebildete Baseballspieler taucht im Spiel nicht auf. Es geht auch nicht um Sport, sondern um acht Stockwerke eines Gebäudes und um viel Streß.

Dieser entsteht immer dann, wenn man die Spielfigur durch die grafisch gut dargestellten Gänge steuert und dabei in alter "Pacman"-Manier Pillen aufsammelt. Im falschen Moment tauchen nämlich merkwürdige Gestalten auf, die sofort angreifen. Man kann sie nur unschädlich machen, wenn man die keulenartigen Gegenstände, die vereinzelt herumliegen, berührt. Erst wenn alle Pillen aufgesammelt sind, kommt man ins nächste Stockwerk. Natürlich wird die Aufgabe hier schwerer.

Obwohl der ganze Spielaufbau "Pacman" gleicht, finde ich "Scooter" doch recht amüsant.

System:	Atari XL/XE
Hersteller:	Americana
Bezugsquelle:	Joysoft

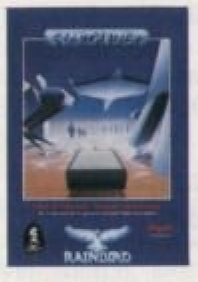
Stephan König

Starglider

Bei diesem Spiel ziehen die Programmierer alle Register ihres Könnens, um den 68000er zu Höchstleistungen zu bringen. Es beginnt mit einigen Sekunden digitalisierter Musik, die bei angeschlossenem Verstärker ihre volle Wirkung entfaltet. Es gibt also für andere Programmierer keine Ausrede mehr, man könnte mit dem Soundchip kaum etwas anfangen; "Starglider" beweist das Gegenteil. Für Spiele sind hervorragende Effekte möglich. Natürlich kann die Musik nicht während des ganzen Spiels ertönen, aber was aus dem Lautsprecher kommt, ist sehr gut.

Auch die Grafik im 3-D-Modus hat mich begeistert. Geschwindigkeit und Auflösung sind faszinierend. Das Cockpit ist ebenfalls hervorragend gelungen. Regler, Anzeigen, blinkende Knöpfe; man fühlt sich wie in einem Science-fiction-Film.

Das Spiel selbst macht durch diese äußeren Anreize natürlich umso mehr Spaß. Nach Start des Programms schlüpfen Sie in die Rolle des Commanders vom modernsten Kampfflugzeugg des Planeten Novenia. Als letzter Überlebender einer geschlagenen Flotte ruhen auf Ihnen alle



1

Hoffnungen, die Aliens doch noch zurückzuschlagen. Wird es ihnen gelingen, sich gegen den übermächtigen Gegner zu wehren? Energie, Munition und Verpflegung sind limitiert, so daß Sie während der nervenaufreibenden Flüge über das besetzte Land immer wieder heimatliche Basen aufsuchen müssen, um nachzuladen. Bei dieser Gelegenheit dürfen Sie ein wenig schlafen, um sich gestärkt in die nächste Schlacht zu stürzen. Vor allem das Landen verlangt sehr viel fliegerisches Können.

Das Flugzeug wird mit der Maus gesteuert, ebenso die Laserkanonen. Die zusätzlichen Funktionen, wie Start, Einschalten des Docking-Computers oder Abfeuern einer Rakete, werden über Tastatur angesprochen.

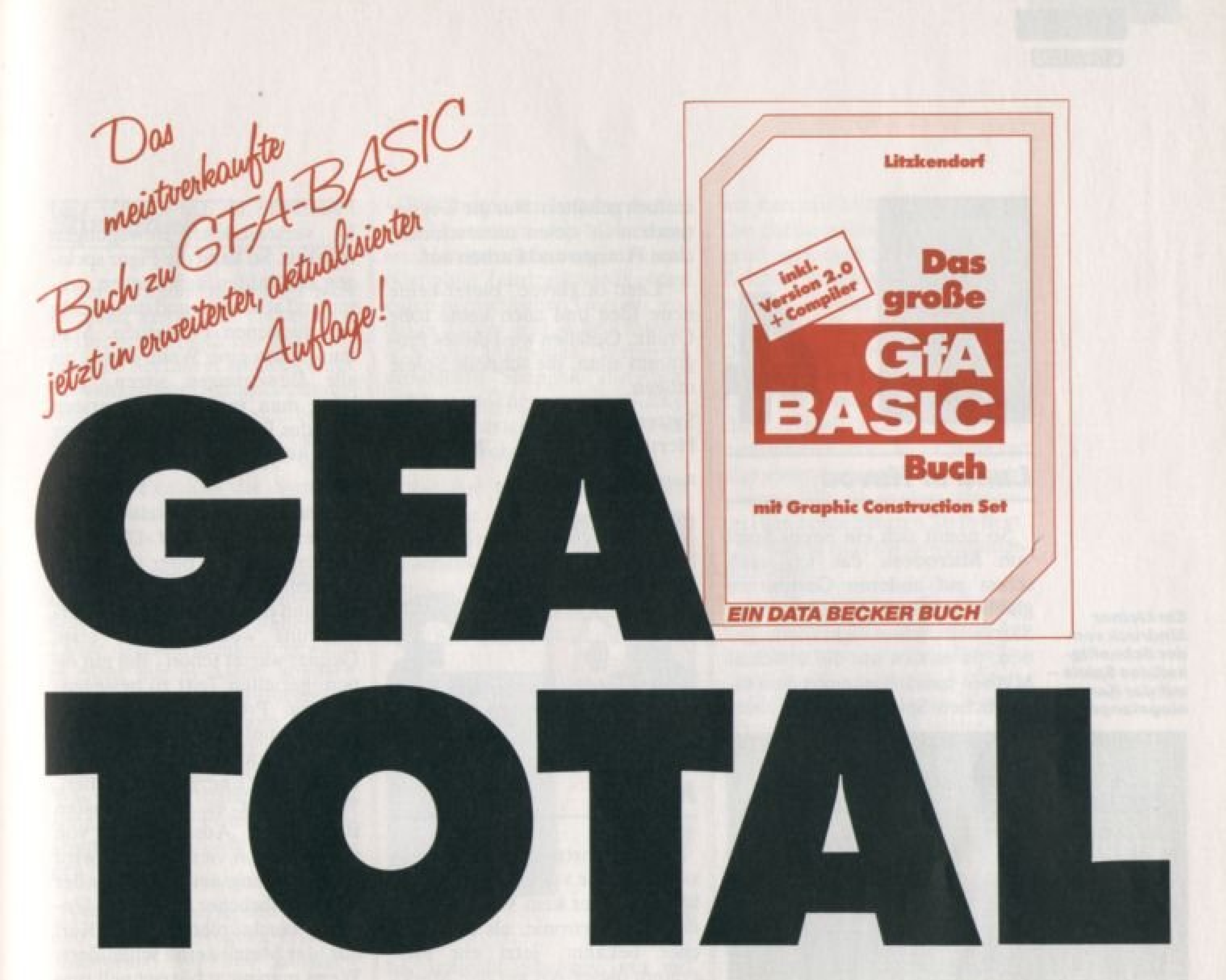
"Starglider" ist eines der besten Spiele, die es zur Zeit für den Atari ST gibt. Es zeigt, was gute Programmierer aus diesem Computer herausholen können.

System:	Atari ST
(512 KByte, SW-/Farbmonitor)	
Hersteller:	Rainbird
Bezugsquelle:	Profisoft

Thomas Kern



"Starglider" zeigt, was bei guter Programmierung aus dem ST herauszuholen ist: eine faszinierende und schnelle Grafik.



Die Idee zum Buch: Anhand eines kompletten, ausbaufähigen Programms sollte der mächtige Befehlssatz des GfA-BASICs vermittelt und erklärt werden. Eine Idee, die den Autor so begeisterte, daß er ein Grafik-Programm der Superlative entwickelte. Im Untertitel bescheiden „Graphic Construction Set" genannt, bietet es alle Standards und zusätzlich zahlreiche, weitere Funktionen: Trickfilm und Animation in Echtzeit, Icon- und Pattern-Editor, Trommelgenerator, 3-D-Animation in atemberaubender Geschwindigkeit, Clipping, Patch-Grafiken... Die Möglichkeiten, die Uwe Litzkendorf hier zeigt, sind fast schon unglaublich. Aber das wichtigste: Nachdem man das Buch gelesen hat, weiß man, wie man sowas programmiert; kennt man alle GfA-Befehle. Denn beim Erstellen des Programms wird jeder neue Befehl detailliert erklärt – und zwar genau dann, wenn er gebraucht wird. Immer eingebunden in kleine Prozeduren, die wiederum in das Programm integriert, aber auch für eigene Programme verwendet werden können. Hierbei lernen Sie dann auch die zahlreichen Tips und Tricks, wie z. B. Rastertechnik, Sprites, System Calls, GEM-Prozeduren, Window- und Objektprogrammierung. Selbstverständlich, daß in diesem Buch auch eine komplette Befehlsübersicht zu finden ist.
Bei solch kompaktem Detailwissen konnte der Erfolg dieses Buches nicht überraschen. Doch dann kam GfA-BASIC 2.0 – das große GfA-BASIC-Buch mußte aktualisiert werden. Kurzerhand ergänzte der Autor das Buch um die neuen Befehle der Version 2.0 und fügte auch gleich dem Programm selbst noch zahlreiche Features hinzu. Ebenfalls in der jetzt erweiterten Auflage: Eine ausführliche Beschreibung des GfA-Compilers und für alle, die einen wirklich einfachen Einstieg suchen, ein detaillierter Einsteigerteil.

Das große GfA-BASIC-Buch
Hardcover
ca. 600 Seiten
DM 49,–

DATA BECKER
Merowingerstr. 30 · 4000 Düsseldorf · Tel. (0211) 310010

BESTELL-COUPON
Einsenden an: DATA BECKER · Merowingerstr. 30 · 4000 Düsseldorf 1
Bitte senden Sie mir:
zzgl. DM 5,– Versandkosten
☐ per Nachnahme ☐ Verrechnungsscheck liegt bei
Name
Straße
Ort



2

International Karate

Von allen Karate-Simulationen, die ich bisher auf dem 8-Bit-Atari gespielt habe, gefällt mir diese Version am besten. Grundidee ist die Austragung eines internationalen Karatewettbewerbs, an dem wahlweise ein oder zwei Spieler teilnehmen können. Das Wort "International" im Programmtitel bezieht sich auf die verschiedenen Hintergrundbilder, die je nach Runde markante Skylines verschiedener Städte wie New York, Sydney usw. zeigen. Diese Grafiken sind teilweise ganz hervorragend gestaltet.

Im Vordergrund treten die beiden Spielfiguren gegeneinander an, wobei die rote auf Wunsch vom Computer übernommen wird. Die Regeln schreiben vor, daß drei Runden mit einer Kampfdauer von jeweils 30 Sekunden auszutragen sind. In jeder Runde können die Kämpfer maximal zwei Punkte erreichen, die von einem Schiedsrichter vergeben werden. Für besonders gelungene Aktionen bekommt man einen vollen, für andere nur einen halben Punkt. Außerdem sind Bonuspunkte möglich, die helfen sollen, einen High Score zu erreichen.

Karate vor der Silhouette Sidneys. Der Computer hat in Form der roten Figur im Augenblick die Oberhand.

Um in ein anderes Bild zu gelangen, muß der Spieler alle drei Runden gewinnen. Andernfalls startet das Spiel wieder von vorne. Die Steuerung erfolgt mit dem Joystick, über den sich 18 verschiedene Schläge, Tritte und Sprünge ausführen lassen. Erreicht wird diese Zahl durch die acht normalen Joystick-Positionen, die durch Druck auf die Feuertaste doppelt belegt werden konnten. Hier eine Auflistung der einzelnen Aktionen:

Springen – Sprungschlag nach vorne – vorwärts gehen – Brusttritt – Fußschlag – Duckschlag – rückwärts gehen – Sprungschlag von hinten – Flugtritt – Überschlag nach vorne – Tritt von vorne – Seitentritt von vorne – Duckschlag von hinten – Seitentritt von hinten – Rundumschlag – Überschlag rückwärts

Am Anfang bereitet die Joystick-Belegung einige Schwierigkeiten; sie erfordert Übung. Schon nach kurzer Zeit kann man aber hervorragend mitspielen und zum Kampf um die Meisterschaft antreten.

Grafik, Animation und Sound wurden bei "International Karate" sehr gut umgesetzt. Auch die Spielmotivation bleibt lange Zeit erhalten, da man ja unbedingt alle Bilder sehen und alle Gegner schlagen will. Wer sich nicht an dem rauhen Kampfsport stört, sollte sich das Programm ansehen. Ich kann es wirklich nur empfehlen.

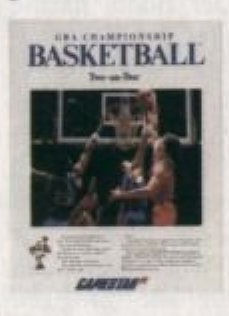
System: Atari 8 Bit
Hersteller: System 3
Bezugsquelle: Activision

Rolf Knorre

Two on Two Basketball Championship

Was sich derzeit auf dem Spielemarkt an Sportsimulationen befindet, ist zumeist keine sehr realitätsnahe Umsetzung der jeweiligen Disziplinen. Vor allem bei Mannschaftssportarten taten sich die Programmierer schwer, denn Taktik, Strategie und vor allem eine gute Koordination unter den Spielern ließen sich nicht so einfach in das Programm einbauen.

"Two on Two" zeigt nun, daß dies doch möglich ist. Nahezu alle Feinheiten, die ein richtiges Basketballspiel ausmachen, findet man hier wieder. Ob man alleine gegen den Computer oder zu zweit spielt, immer fühlt man sich mitten aufs Spielfeld versetzt. Um vor allem die Teamarbeit so gut wie möglich zu simulieren, verzichtete man auf die volle Mannschaftszahl; es spielen immer Zweierteams gegeneinander.



3

In verschiedenen Modi kann der einzelne Spieler das Treffen des Korbes üben oder direkt ins Geschehen eingreifen, indem er mit einem Computerpartner gegen eine Computermannschaft antritt. Zwei Spieler können jeweils in gegnerischen Mannschaften spielen und den Computer zum Partner nehmen oder sich auch gegen ihn verbünden, was wohl die reizvollste Version darstellt. Eine Mannschaft wird mit zwei Joysticks gesteuert, und es kommen wirkliche Spielzüge zustande. Natürlich beherrscht auch der Computer die Kunst des Dribbelns und Treffens, doch mit ein wenig Teamgeist kann man zu zweit schon einen Sieg gegen die Technik erringen.

Ziel des Spiels ist es nun, den Basketball Championship zu gewinnen. Dazu muß gegen eine Reihe von Mannschaften in einer von vier Divisionen gespielt werden. Den Gegner sucht jeweils



Spannende Sportsimulation mit kleinen Gags: Basketball auf dem ST.

der Computer aus, und anhand einer Tabelle im Handbuch kann man seine Stärke herausfinden und die eigene Mannschaft darauf einstellen.

"Two on Two Basketball Championship" eröffnet neue Möglichkeiten der Sportsimulation auf dem Computer, die bisher einmalig sind. Leider ist die Umsetzung auf den Atari ST nicht ganz so gelungen, wie man es hätte erwarten können. Die Figuren sind relativ klein, unscheinbar und lieblos gezeichnet. Ein wenig mehr Mühe bei der Gestaltung der Animation hätte bestimmt nicht geschadet. Dennoch fällt die Gesamtwertung sehr gut aus, denn an Spielspaß und Qualität gibt es nichts Vergleichbares. Kleine Gags lockern das Ganze noch zusätzlich auf. So erscheint immer vor und nach einem Spiel die entsprechende Seite eines Sportmagazins mit einem Bericht über das Ereignis. Wer also Wert auf eine gute, spannende und abwechslungsreiche Simulation legt, ist mit diesem Programm richtig beraten.

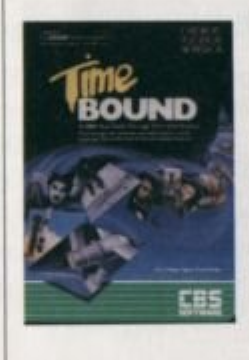
System: Atari ST 512 KByte, Farbmonitor
Hersteller: Gamestar/Activision
Bezugsquelle: Profisoft

Thomas Tai

Timebound

Etwas aus dem Rahmen fällt das Programm "Timebound"; es wendet sich an die jüngeren bis ganz jungen Computerfreunde und versucht, neben dem Spielwert auch noch etwas Allgemeinwissen zu vermitteln. Um den Spieltrieb zu wecken, wurde das Programm natürlich mit einer passenden Geschichte versehen.

Der arme Anachron hat sich entgegen allen Warnungen mit einer Zeitmaschine beschäftigt und hängt jetzt in einer Zeitschleife fest, in der er ständig hin und her katapultiert wird. Sie umfaßt 2000 Jahre, beginnend



3

mit dem Jahr 0. Der Spieler (das Programm ist nur für einen Teilnehmer ausgelegt) muß versuchen, Anachron zu retten. Als Hilfsmittel steht ihm ein Scanner zur Verfügung, mit dem er ebenfalls durch die Zeit reisen kann. Leider sind keine gezielten Sprünge möglich. Man kann nur vor- oder zurückfahren.

Auf dem Bildschirm erscheinen ständig rote Punkte. Hat man einen davon mit dem Scanner eingefangen, stoppt die Zeitreise. Im oberen Bildschirmbereich stehen nun die aktuelle Jahreszahl und ein Begriff. Dieser soll ein herausragendes Ereignis bezeichnen, das im angezeigten Jahr stattgefunden hat (Beispiel: Jahr = 1960 – Begriff = Laser). Am unteren Bildschirmrand erscheint ein weiterer Begriff, der einen Hinweis auf Anachrons momentanen Aufenthaltsort gibt. Dort ist jedoch keine Jahreszahl vermerkt.

Um den Zeitreisenden zu finden, muß man also immer darauf achten, ob der angegebene Begriff schon einmal im oberen Bildschirmbereich auftauchte, und sich daran erinnern, mit welcher Jahreszahl er dort verbunden war. Befindet sich der Spieler z.B. gerade irgendwo im Mittelalter, und der Begriff Laser erscheint, gilt es, schnellstmöglich in das Jahr 1960 zu gelangen. Mit einem guten Gedächtnis und etwas Glück kommt man so bald ans Ziel. Allerdings bleibt Anachron ja nicht unbegrenzt in dem betreffenden Jahr.

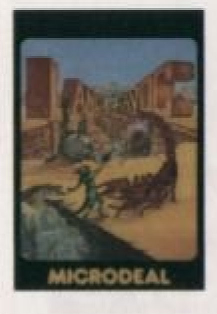
Da die Kommentare in Englisch gehalten sind, wird das Programm bei uns wohl nicht den Lerneffekt haben können, den sich die Programmierer erhofft haben. Auf jeden Fall werden jüngere Spieler nicht allzu viel damit anzufangen wissen. Ansonsten handelt es sich bei "Timebound" um ein lustiges Spiel mit günstigem Nebeneffekt.

System: Atari 8 Bit
Hersteller: CBS Software
Bezugsquelle: Diabolo

Stephan König

Zeitreisen mit Lerneffekt. Mit "Timebound" wird das Pauken von Jahreszahlen zum Spiel.



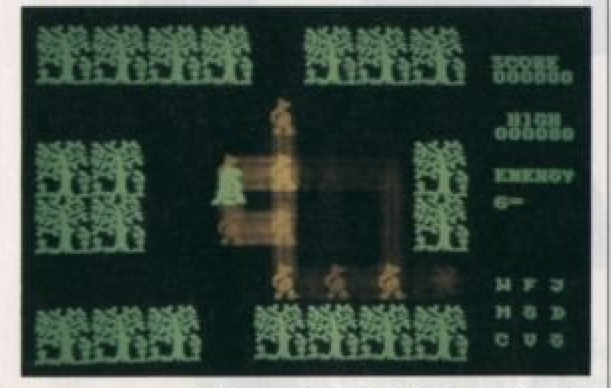


4

Land of Havoc

So nennt sich ein neues Spiel von Microdeal, das ich auch schon auf anderen Computern glaube gesehen zu haben. Die Story will ich hier nicht wiedergeben, da es sich um die üblichen Mythen handelt, die mit dem eigentlichen Spielgeschehen überhaupt nichts zu tun haben. Im Grunde handelt es sich um ein Labyrinthspiel mit angeblich 2000 Räumen. (Ich habe sie nicht nachgezählt.)

Ein kleiner Eindruck von der Schnelligkeit des Spiels – mit der Kamera eingefangen.



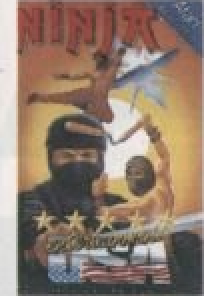
Hauptdarsteller ist ein kleiner, saurierähnlicher, grüner Kerl. Er soll das Land Havoc erkunden und befreien. Auffälligstes Merkmal dieses Programms ist die Geschwindigkeit, mit der man die Figur durch die Räume jagen kann. Ähnlich schnell tauchen auch die zahlreichen Gegner auf. Wenn man nicht ganz fix ist, hat man die 6 Leben, die zur Verfügung stehen, in einigen Sekunden verbraucht.

Zur Orientierung liegen der Diskette einige Karten bei, die entsprechend einer Vorgabe zusammengelegt werden sollen. Mit dieser Hilfe weiß man wenigstens ungefähr, wo man sich gerade befindet. Die Grafik ist eher einfach gehalten. Nur die Gegner tauchen in vielen unterschiedlichen Formen und Farben auf.

"Land of Havoc" bietet keine neue Idee und auch keine tolle Grafik. Gefallen wird dieses Programm allen, die schnelle Spiele mögen.

System: Atari ST
Hersteller: Microdeal

Rolf Knorre



3

Ninja

Kampfsportspiele erfreuen sich nach wie vor großer Beliebtheit. So ist es kein Wunder, daß auch Mastertronic, als Preisbrecher bekannt, jetzt ein Programm dieser Art auf den Markt gebracht hat. Es nennt sich "Ninja" und soll laut Werbung alle anderen Programme ausstechen.

Ninja ist ein harter Kämpfer in schwarzem Gewand, der sich auf der Suche nach der gefangenen Prinzessin Di-Di, der Perle des Orients, befindet. Diese wird natürlich schwer bewacht.

Die Steuerung des Kämpfers ist nicht ganz einfach, da der Joystick mit zahlreichen Funktionen belegt ist. In Verbindung mit der Feuertaste als Umschalter sind 14 verschiedene Bewegungen möglich. So kann die Figur springen und laufen,sich ducken, treten und schlagen, und das alles in verschiedenen Positionen. Man muß schon eine Weile üben, bis alle Bewegungen sitzen. Das kann man recht gut im ersten Bild des Programms tun, da hier noch kein Gegner zu sehen ist.

Der Ablauf des Spiels ist nicht sonderlich aufregend. Die Figur muß nach rechts oder links zum Bildschirmrand geführt werden, wo dann das Bild mit kurzer Verzögerung wechselt. Der erste Gegner wartet schon . Ihn mit einem gezielten Tritt zu besiegen, ist kein Problem. Wenn nach dem Kampf ein blinkendes Etwas am Boden liegt, kann man damit seine Energie auffrischen. Weiter geht es zum nächsten Bild. Nach Ausschaltung von drei oder von vier Gegnern wird die Sache langsam anstrengender bzw. gefährlicher. Zu allem Unglück taucht plötzlich ein Kerl auf, der Metallsterne schleudert. Wenn man nicht blitzschnell reagiert, ist das Spielende erreicht.

"Ninja" ist ein unterhaltsames Spiel, wenn man von der recht brutalen Handlung einmal absieht. Allerdings ist es in den meisten Fällen nicht besonders schwierig, den Gegner zu besiegen.

System: Atari 800 XL / 130 XE
Hersteller: Mastertronic
Bezugsquelle: Compy Shop

Stephan König

"Ninja" – ein Kampfsport aus dem fernen Osten. Mit dem Kämpfer können Sie 14 verschiedene Bewegungen vollziehen.



Shanghai

Vor etwa 2000 Jahren, als noch niemand etwas von Computern ahnte, setzten sich ein paar chinesische Seefahrer an Deck einer schwankenden Dschunke zusammen und vergnügten sich bei einem seltsamen Spiel mit bemalten Steinchen. Später ließen reichere Familien die Symbole auf rechteckig geschliffene Elfenbeinstückchen malen.

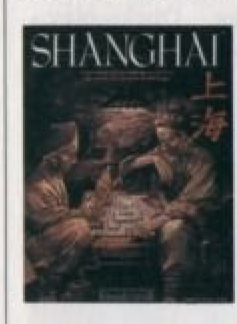
So oder ähnlich kann man sich die Entstehung des traditionellen Spiels Mah Jongg vorstellen. Im Laufe der Jahrhunderte gewann es nicht nur in Asien, sondern auch in Amerika und Europa an Beliebtheit. Der Boom flachte aber im Zeitalter des elektronischen Spielzeugs deutlich ab. Was liegt nun näher, als Mah Jongg selbst in Form eines Programms auf den Markt zu bringen?

Activision wagte den Schritt, und in diesen Wochen erschien "Shanghai" für alle gängigen Computersysteme. Dabei handelt es sich um eine leicht geänderte Version des traditionellen Mah Jongg, doch die Faszination dieses Spiels ging nicht verloren.

Bevor wir uns die Umsetzung auf den Atari ST ein wenig genauer ansehen, sollen kurz die Regeln erläutert werden. Das Spiel besteht aus 144 Steinen mit Symbolen verschiedener Art; dabei ist jedes Motiv viermal vorhanden, und es gilt nun, jeweils ein passendes Paar zu finden. Zu Beginn sind alle Steine kunstvoll zu der Figur eines sog. Drachens aufgetürmt, d.h. sie liegen in mehreren Schichten übereinander und dicht bei dicht. Bewegt werden dürfen nur freie Steine. Darunter versteht man solche, die sich nach rechts oder links wegnehmen lassen. Aus der Menge der freien Steine müssen passende Paare herausgefunden werden, die man dann entfernen darf. Es gilt nun, auf diese Art den gesamten Berg abzutragen.

Das Programm bietet verschiedene Versionen von Mah Jongg an. Man kann alleine spielen, mit oder ohne Zeitdruck sowie gegen einen menschlichen Gegner, wobei es darum geht, möglichst viele Steine abzubauen. Weiterhin ermöglicht "Shanghai" die Rücknahme von Zügen und die Analyse des bisherigen Spiels, um Fehler aufzudecken.

Bei der Umsetzung auf den Computer war es besonders wichtig, Struktur und Aufbau des Drachens deutlich darzustellen, denn sonst läßt sich der Status der einzelnen Steine nicht erkennen. Auf dem C 64 ist dies denk-



2

bar schlecht gelungen; nur mit viel Fantasie findet man passende Steine. Anders verhält es sich beim Amiga; sowohl die Symbole als auch der dreidimensionale Effekt wurden hervorragend realisiert. Mit der ST-Version haben wir nun ein Mittelding vor uns. Die Steine selbst sind hervorragend gelungen; Symbole und Zeichen lassen sich sehr gut erkennen. Leider gilt dies aber nicht für die dreidimensionale Darstellung. Um die einzelnen Schichten des Drachens und die Lage der Steine deutlich zu machen, erhielt jede Schicht eine bestimmte Farbe. Jeder Stein hat also einen farbigen Rand, der erkennen läßt, ob er über oder unter dem benachbarten Stein liegt. Das Ganze ist recht gewöhnungsbedürftig.

Die Faszination, die "Shanghai" ausübt, läßt sich kaum begreifen, wenn man es nicht selbst einmal ausprobiert hat. An Spielspaß für kalte und lange Winterabende mangelt es dann bestimmt nicht mehr. Vor allem, wenn zwei Spieler gegeneinander antreten, werden die Köpfe schon mal heiß. Nach einiger Zeit hat man sich auch an die zweidimensionale Darstellung gewöhnt und spielt am Computer, als hätte man einen richtigen Berg von Mah-Jongg-Steinen vor sich.

System: Atari ST 512 KByte, Farbmonitor
Hersteller: Activision
Bezugsquelle: Profisoft

Thomas Tai

Bei "Shanghai" wird der Spieler im Vergleich zu Memory weitaus mehr gefordert.



INSERENTEN

AMC-Verlag
86
Buchversand
66, 67
CCD
84, 85
Computer Kontakt
21
Computer Service
6
Compy Shop
39
Copy Data
33
Data Becker
23, 79, 99
Diabolo-Versand
93
Dörr
7
Engl Computertechnik
3
GFA
100
Holschuh
7
Idee-Soft
80
Irata-Verlag
9, 36
Kull
6
Lindenschmidt
24
New's
85
Omikron
81
Padercomp
36
PC-Software
87
Rätz-Eberle
73
Rausch & Haub
9
RSE Schuster
49
Rudolph
3
tewi-Verlag
2

Anzeigenschluß für die Februar/März-Ausgabe ist der 30.3.1987.

Falls Fragen auftauchen, steht Ihnen Herr Arno Weiß gerne zur Verfügung.
☎ 07252/3058

VORSCHAU

GFA

Das deutsche Softwarehaus GFA Systemtechnik hat sich bei vielen ST-Besitzern bereits einen guten Namen durch das GFA-Basic gemacht. Es gibt mittlerweile aber nicht nur diesen Basic-Interpreter, sondern auch andere Programme mit der Vorsilbe GFA. Die Rede ist von GFA-DRAFT, GFA-VEKTOR und natürlich dem lange erwarteten GFA-Basic-Compiler, der das ohnehin schon schnelle Basic aus Düsseldorf noch flotter machen soll. Lesen Sie dazu die Testberichte im nächsten **ATARI** *magazin*

Plotter

Viele Computerbesitzer verfügen bereits über einen Drucker. Weit weniger verbreitet sind die sogenannten Plotter, die im CAD/CAM-Bereich eine wichtige Rolle spielen. Wir haben für Sie den Low-Cost-Plotter der Firma OPUS getestet, der mit rund 800.– DM auch für Heimanwender erschwinglich ist.

Hardcopy

Wer einen Atari 800 XL / 130 XE und einen Matrixdrucker besitzt, will damit neben Textverarbeitung und anderen Anwendungen sicher auch Grafiken zu Papier bringen. Ein neues Programm soll dazu eine Hilfestellung bieten. Was es leistet, können Sie im nächsten **ATARI** *magazin* lesen.

Spiele

Ohne Zweifel gehören die "kleinen" Atari-Computer noch immer zu den beliebtesten Einsteigermodellen. Das beweisen auch die ständig neu erscheinenden Spiele für diese Rechner. "Asylum", das "Boulder Dash Construction Kit", "Masterchess", "Despatch Rider" und andere haben wir unter die Lupe genommen.

Das nächste ATARI *magazin* erscheint am 22.4.87

IMPRESSUM

Herausgeber
Werner Rätz
Dipl.-Wirt.-Ing. (FH) Thomas Eberle

Chefredakteur
Thomas Eberle

Technische Redaktion
Werner Rätz

Redaktion
Helmut Fischer
Robert Kaltenbrunn

Ständige freie Mitarbeiter
Rolf Knorre
Thomas Tai
Dipl.-Ing. Peter Finzel
Thomas Tausend

Versandservice
Gabriele Herzog

Titelbild
Rainer Grinda

Anzeigen
Arno Weiß
Es gilt die Anzeigenpreisliste Nr. 1

Montage
Frederique Melchers

Satz
Druckerei Sprenger
7143 Vaihingen/Enz

Druck
Südd. Zeitungsdienst GmbH
Druckerei- u. Verlags-GmbH
7080 Aalen

Vertrieb
Verlagsunion
6200 Wiesbaden

Anschrift des Verlags
Verlag Rätz-Eberle
Postfach 1640
Melanchthonstraße 75/1
7518 Bretten
Telefon 0 72 52 / 30 58

Das **ATARI** *magazin* erscheint zweimonatlich jeweils zur Mitte des Vormonats.
Das Einzelheft kostet 6.– DM.



DIABOLO

★ Der Versand mit den teuflischen Preisen! ★

Atari 8 Bit

	Cass.	Disk.
Action Biker	DM 9.90	—.—
Airwolf	DM 25.90	—.—
Asylum	DM 25.90	39.90
Bilbo	DM —.—	19.—
BMX Simulator	DM 14.90	—.—
Boulder Dash Construction Set	DM 25.90	39.90
Crystal Rider	DM 9.90	—.—
Datestone of Ryn	DM 12.90	—.—
Designmaster	DM —.—	19.80
Despatch Rider	DM 9.90	—.—
Fighter Pilot	DM 25.90	39.90
Ghostbusters	DM 25.90	39.90
Goonies	DM 25.90	39.90
Gun Law	DM 9.90	—.—
Hacker	DM 25.90	39.90
International Karate	DM 25.90	39.90
Jump Jet	DM 25.90	39.90
Kik Start	DM 9.90	—.—
King Size (50 Spiele)	DM 29.90	—.—
Koronis Rift	DM 25.90	39.90
Last V 8	DM 14.90	—.—
Leaderboard	DM 25.90	39.90
Los Angeles Sweat	DM 14.90	—.—
Master Chess	DM 9.90	—.—
Mike's Slotmachine	DM —.—	19.—
Molecule Man	DM 9.90	—.—
Ninja	DM 14.90	—.—
One Man and his Droid	DM 9.90	—.—
Polar Pierre	DM 25.90	39.90
Pyramidos	DM —.—	29.—
Red Max	DM 14.90	—.—
Rescue on Fractalus	DM 25.90	39.90
Spellbound	DM 14.90	—.—
Spy vs Spy II	DM 25.90	39.90
Tales of Dragons	DM —.—	19.—
The Soundmachine	DM —.—	29.80
s.w.a.t.	DM 9.90	—.—
Tomahawk	DM 25.90	39.90
Vegas Jack Pot	DM 9.90	—.—
Zorro	DM 25.90	39.90



ST-Power

	Disk.
Bratacass	DM 49.—
Land of Havoc	DM 49.—
Major Motion	DM 49.—
Mercenary	DM 49.—
Space Station	DM 49.—
ST Karate	DM 49.—
Time Bandit	DM 49.—

Software-Bestellschein

Ich bestelle aus dem Diabolo-Versand folgende Software:

Anzahl	Titel	Gesamtpreis

Ich wünsche folgende Bezahlung:

☐ Nachnahme (zuzüglich 5.70 DM Versandkosten)

☐ Vorauskasse (zuzüglich 3 DM Versandkosten, ab 100 DM Bestellwert versandkostenfrei)

Bei Vorauskasse bitte Scheck beilegen

Ⓐ

Name des Bestellers

Anschrift

PLZ/Ort

Datum/Unterschrift

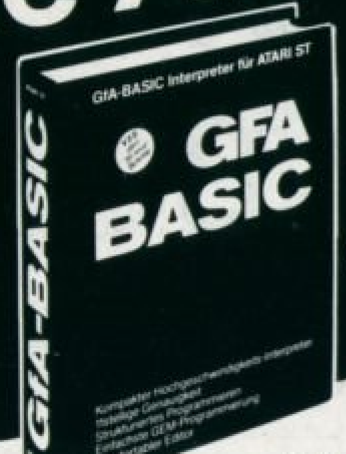
Coupon ausschneiden, auf Postkarte kleben

und einsenden an:

Diabolo-Versand, Postfach 16 40, 7518 Bretten.

Eine Abteilung des Verlags Rätz-Eberle GdbR.

Für alle ATARI ST



Kompakter Hochgeschwindigkeits - Interpreter,
11stellige Genauigkeit,
strukturiertes Programmieren,
einfachste GEM-Programmierung,
komfortabler Editor.

GFA-BASIC Interpreter V 2.0
DM 169,–

GFA-BASIC Compiler für ATARI ST
GFA
BASIC
GFA-BASIC

GFA-BASIC Compiler
DM 169,–

Schneller 2-Pass-Compiler, benötigt keinen Linker,
erzeugt sehr schnelle, kurze Programme ohne Runtime-Modul,
voll kompatibel zum Interpreter, einfachste Bedienung.



GFA-Buch DM 79,–

Incl. Diskette mit Beispiel-Programmen
Lieferbar voraussichtlich Ende Februar 87

3D-Animationsprogramm für ATARI ST
GFA
VEKTOR
GFA-VEKTOR

Schnelles, interaktives 3D-Grafik-Programm zum Generieren von 2D- und 3D-Objekten.

Aus GFA-BASIC heraus können problemlos 2D- oder 3D-Spiele, Animationen oder bewegte Simulationen erzeugt werden.

GFA-VEKTOR DM 149,–

CAD-Programm für ATARI ST
GFA
DRAFT
GFA-DRAFT

Voll GEM-gesteuertes, leistungsfähiges CAD-Programm, maßstabsgerechtes Erstellen von Zeichnungen in Zoll und mm, Bildausschnitte und Symbole beliebig manipulierbar und gradweise drehbar, mächtige Zeichenfunktionen wie z. B. Lot, Winkel zu Geraden, Kreis durch 3 Punkte, etc. 255 Zeichenebenen (10 gleichzeitig darstellbar). Schraffieren von beliebigen Flächen. Einfachste Handhabung. Läuft auf fast allen Plottern und Druckern. Symbole und Bibliotheken in beliebiger Menge anlegbar.

GFA-DRAFT DM 298,–

...Anruf genügt: 02 11-58 80 11

GFA Systemtechnik GmbH

Heerdter Sandberg 30
D-4000 Düsseldorf 11
Telefon 02 11/58 80 11

GFA
SYSTEMTECHNIK